UNIVERSIDAD CARLOS III DE MADRID

TESIS DOCTORAL

INGENIERÍA MATEMÁTICA

Combinación y suavizado de series de tiempo para el análisis demográfico

Autor:
José Eliud Silva Urrutia

Director/es:
Daniel Peña

Víctor M. Guerrero[1]

DEPARTAMENTO DE ESTADÍSTICA

Getafe, febrero 2010

[1]Departamento de Estadística. Instituto Tecnológico Autónomo de México (ITAM), 01000 D. F. México, México.

TESIS DOCTORAL

Combinación y suavizado de series de tiempo para el análisis demográfico

Autor:

José Eliud Silva Urrutia

Director/es:

Daniel Peña/Víctor M. Guerrero

Firma del Tribunal Calificador:

		Firma
Presidente:	Antoni Espasa	
Vocal:	Maria Jesus Sanchez	
Vocal:	Ana Justel	
Vocal:	Pilar Poncela	
Secretario:	Andrés M. Alonso Fernández	

Calificación:

Getafe, febrero de 2010

Del mito

Mi madre me contó que yo lloré en su vientre. A ella le dijeron: tendrá suerte.

Alguien me habló todos los días de mi vida al oído, despacio, lentamente.

Me dijo: ¡Vive, vive, vive!. Era la muerte.

Jaime Sabines-Poeta mexicano

(1926-1999)

…Y al final, seguía un Ángel fiel con su amistad…

Agradecimientos

Quiero dar las gracias a mis Directores de tesis. A ambos por su gran apoyo, enseñanza, confianza, motivación, paciencia y por darme la oportunidad de trabajar con ellos. Asimismo, al Dr. Víctor M. Guerrero por difundir algunos hallazgos de esta tesis en congresos nacionales e internacionales, y, al Dr. Daniel Peña por invitarme a participar en su proyecto MICINN Grant SEJ2007-64500.

A la Universidad Carlos III de Madrid y al Departamento de Estadística por confiar en un estudiante extranjero con deseos de superación. Un agradecimiento a mis amigos que me apoyaron en todo momento en España. Así como también a los profesores que me brindaron sus conocimientos. En México a mis amigos del CENEVAL, UNAM, El Colegio de México, CONACULTA y la Universidad Virtual Liverpool. A todos los alumnos que he tenido durante mi labor docente, por ser fuente de inspiración para seguir adelante.

Muy especialmente a mi madre, María Guadalupe, por su enseñanza en el control de la estabilidad emocional ante cualquier circunstancia; a mi padre, José Guadalupe, por su sonrisa fresca y su sutil ejemplo; a Claudia por tanta dulzura y alegría ante avances de terceros; a Gabriela por su muestra de familia posmoderna y buen humor; a Lalo por su innegable éxito tras éxito en todo y atreverse a *ser*; a Manuelo por su notable apego familiar; a Lisa por su gran respeto a mi persona; a Andresito por compartir todo conmigo; a Zita, Gonzálo y Bob por su complicidad, crecer juntos y su infinita comprensión. Finalmente y con mucho cariño a mi país México. Sinceramente no sé como pueda agradecerles todo lo que han hecho por mí…

Contents

Chapter 0. Resumen y conclusiones principales

Las técnicas y procedimientos de la estadística se pueden aplicar para la comprensión y resolución de problemas en diversas áreas del conocimiento. En la demografía, en particular en el Análisis Demográfico, se tiene una gran beta de oportunidad para su aplicación donde, en los últimos años, gran parte de su desarrollo se ha sustentado en la aplicación y surgimiento de propuestas metodológicas, no tan solo elaborados *ex profeso* en la demografía sino generalmente en otras disciplinas. Principalmente desde la década de los 80´s se han suscitado investigaciones en el campo demográfico, bajo la óptica de las series de tiempo, donde se ha abordado las distintas componentes que influyen en la evolución de las poblaciones: fecundidad, mortalidad y migración. Dichas investigaciones tienen como común denominador el abocarse al modelaje, la descripción, el análisis y el pronóstico.

Para la fecundidad se tienen entre otros, por ejemplo, los trabajos de: Land y Cantor (1983); Carter y Lee (1986); De Beer (1989); Thompson et al. (1989); De Beer (1992); Lee (1993); Rogers et al. (1993); Bell (1997); Durban et al. (2001); McNown y Rajbhandary (2003); McNown y Ridao-Cano (2005) y Jeon y Shields (2008). Por otro lado, para la mortalidad destacan los trabajos de: McNown y Rogers (1989a, 1989b y 1992); Laporte y Fergusonb (2003); Alonso (2008); Goldstein (2009). Asimismo para la migración, rama de la demografía mucho menos explotada: Brücker et al. (2003) y Cornwell (2009). Finalmente, para pronósticos de población se tienen, entre varios: De Beer (1985); Lee (1992); Lee y Tuljapurkar (1994); Keilman et al. (2002); Girosi y King (2004); Tuljapurkar et al. (2004); Hyndman y Booth (2008); Alonso et al. (2009) y Okita et al. (2009).

La investigación que se presenta en esta tesis se constituye por tres trabajos, donde se establecen temas desde distintas posibilidades reales y que frecuentemente pueden aparecer en el ámbito del quehacer demográfico o actuarial interactuante con la estadística. Esto se aborda por medio de aplicaciones y propuestas metodológicas, donde se emplean series temporales univariantes y multivariantes de tipo demográfico. Se tiene certeza de que dichas propuestas aportan estrategias de descripción y análisis extendibles a otros campos, y que incluso en la misma demografía se pueden seguir desarrollando nuevas líneas de investigación con base a lo aquí expuesto. En breve, en la tesis se exploran, exponen y proponen tópicos de combinación de información demográfica, suavizamiento de tendencias de series de mortalidad y su control a través del criterio del analista, así como la conjunción de ambas tareas simultáneamente.

Se está convencido de que las propuestas metodológicas, ilustradas con ejemplos prácticos demográficos, abonan a la frontera conceptual entre el Análisis Demográfico y las series temporales, tanto en el caso univariante como en el multivariante. En los capítulos de la tesis, en su caso, existen proposiciones demostradas, descripciones detalladas de puntos específicos, referencia de los datos utilizados y el conjunto de programas de cómputo utilizados, todo ello ubicado en los apéndices respectivos. En este sentido, queda de manifiesto la posibilidad de utilizar diversos programas de cómputo (ya sean estadísticos, econométricos o matemáticos), cuya existencia en el mercado no ha sido consecuencia directa de dar soluciones a los problemas del Análisis Demográfico. Entre otros y de los aquí empleados se tienen: E-Views y RATS o Matlab y R.

A continuación se detallan los contenidos de los capítulos centrales de la tesis, se enmarcan algunas conclusiones relevantes de los mismos y finalmente se sugieren algunas líneas de investigación futuras.

En el primer trabajo (Chapter 2), se muestran algunas aplicaciones de métodos de series de tiempo para resolver dos problemas típicos que surgen de manera recurrente cuando se analiza información demográfica en países subdesarrollados o en regiones donde no hay un registro demográfico recurrente. A saber: (1) la falta de existencia de series anuales de los niveles de la población o sus crecimientos anuales y (2) la falta de estrategias apropiadas para definir las metas de crecimiento demográfico dentro de programas oficiales de población, con base en su propio registro histórico. Ambos problemas, se consideran dentro de la tesis como situaciones donde se requiere la combinación de información de series de tiempo de población. En primer lugar, se sugiere la utilización de las denominadas técnicas de desagregación temporal para combinar los datos decenales de distintas ediciones de ejercicios censales con información anual de estadísticas vitales, a fin de estimar las tasas de crecimiento anual de la población. Se opta por utilizar la propuesta realizada por Guerrero y Nieto (1999), puesto que se considera la más apropiada por su característica de sustentarse en no asumir estructuras específicas de los errores aleatorios involucrados, sino más bien tomar en cuenta los rasgos particulares de los datos bajo estudio (que en este caso son estrictamente demográficos).

Posteriormente, una vez desagregadas las series e incorporando una medida de error de variabilidad derivado de la desagregación, se aplica la técnica de pronósticos restringidos múltiples, para combinar las metas oficiales de los futuros índices de crecimiento de la población, siguiendo la idea de Pankratz (1989). Entonces, se propone un mecanismo para evaluar la compatibilidad de los objetivos demográficos con los datos anuales, utilizando para este fin las pruebas estadísticas de compatibilidad de Guerrero y Peña (2000, 2003). Se aplican los procedimientos antes mencionados a los datos de la Zona Metropolitana de la Ciudad México (ZMCM) dividida por anillos concéntricos de desarrollo urbano, los cuales están conformados por unidades geográficas llamadas municipios y delegaciones.

Entre varias conclusiones, se verifica que los objetivos establecidos en el programa oficial no son factibles de alcanzar, siendo una mera aspiración sin un solido respaldo en la dinámica demográfica de la ZMCM. También, este análisis indica que antes de proponer metas demográficas, es muy recomendable evaluar su viabilidad empírica y objetiva. Por lo tanto, se proponen tasas futuras de crecimiento de población que están dentro de la región de factibilidad en consonancia con el comportamiento demográfico histórico. Por último, se concluye que las estrategias metodológicas presentadas pueden ser utilizadas además de en países en desarrollo, en otras regiones geográficas donde existan dichos problemas. Igualmente, se considera que los programas de crecimiento de la población podrían establecerse dando un seguimiento minucioso con este tipo de análisis. Para este trabajo se utilizan los softwares: E-Views versión 5 y Matlab versión 7.

En el segundo trabajo (Chapter 3), se propone un método que permite estimar tendencias de mortalidad en series tiempo, donde se emplean los denominados B-splines (Eilers y Marx, 1996). Esto se propone de forma que el usuario pueda fijar un porcentaje de suavidad deseado, con lo que se logra la comparabilidad de tendencias con iguales porcentajes de suavidad. También con esta propuesta se prevé que es posible estimar datos faltantes o realizar pronósticos de manera relativamente sencilla. Se introduce la importancia del método aplicado a tasas de mortalidad para el diagnóstico y la toma de decisiones dentro del sector asegurador o en el marco del diseño de políticas de población. La idea de comparabilidad de tendencias o tasas de mortalidad suavizadas se desarrolla a partir del cálculo de un índice de suavidad cuyas propiedades se hacen explicitas y se demuestran.

Se exponen algunos resultados teóricos en el suavizamiento tanto en el caso unidimensional como en el bidimensional, y, se proponen índices de suavidad, siguiendo y generalizando la idea de Guerrero (2008). Se observa que es factible identificar entre otras, la relación existente entre el índice de suavidad unidimensional con el respectivo en el espacio bidimensional; también, el comportamiento de los índices respecto a sus cotas cuando algunos parámetros se asumen en determinada dirección. Es importante señalar que en esta propuesta, se generan resultados pioneros en el ámbito del suavizamiento bidimensional desde otra óptica, la cual tiene como valor agregado permitir al usuario elegir un porcentaje de suavidad acorde a su experiencia en la materia. Cabe notar que no se sugiere de ninguna manera dar un giro radical sobre lo existente en el tema o descalificarlo por esta propuesta, sino más bien, solo se busca dar una óptica distinta al problema del suavizamiento.

Los resultados obtenidos tienen un sustento sólido matemático-estadístico, y en particular, se demuestra la equivalencia que existe entre los planteamientos conocidos con los propuestos al usar Mínimos Cuadrados Generalizados (MCG). Los cálculos pueden realizarse de manera eficiente sin que sea necesario invertir matrices de altas dimensiones. Esto es gracias a que se emplean resultados documentados en la literatura (Ruppert, 2002). Se presentan ejemplos con datos del Continuous Mortality Investigation Bureau del Reino Unido para edades de 11-100 y para los años 1947-1999, ya utilizados con antelación (Currie y Durban, 2002), que permiten apreciar y contrastar el tipo de resultados que se pueden obtener al aplicar la metodología propuesta. Para este trabajo se utilizan los softwares R (R-2.6.2) y Matlab versión 7.

En el tercer trabajo (Chapter 4), se realiza una propuesta metodológica que resulta útil para estimar tendencias en series de mortalidad al considerar ajuste, suavidad e información proveniente de una estructura de mortalidad dada desde una óptica no paramétrica. Una de las ventajas más notables de dicha metodología es la posibilidad de que el analista dé mayor, menor o igual credibilidad a una fuente de información sobre otra. Asimismo permite que el analista controle un porcentaje de suavidad y estructura de acuerdo a sus intereses, con la finalidad de lograr comparabilidad. De alguna manera, se da seguimiento a la definición de índices de suavidad propuestos en el trabajo previo, también bajo la idea de Guerrero (2008).

Cabe destacar que algunas circunstancias que podrían presentarse al aplicar esta propuesta metodológica podrían ser: la presencia de datos faltantes o que las fuentes de información tengan distinto tamaño (es decir, que por ejemplo una experiencia de mortalidad dada tenga mayor cantidad de datos para más edades en relación con otra experiencia de mortalidad). Sin embargo, ambas situaciones se superan a través de la utilización del llamado Filtro de Kalman.

Dentro de los ejemplos se emplean datos de mortalidad de Japón, Inglaterra, Chile, Estados Unidos y México. Provienen principalmente del sitio www.mortality.org/, The Human Mortality Database (HMD), apoyado por la Universidad de Berkley y el Instituto Max Planck para la investigación demográfica. En todos los casos, se considera que los resultados son convincentes de acuerdo a la lógica demográfica. Finalmente, cabe notar que la aplicación de la metodología puede realizarse sobre otros tipos de indicadores demográficos de mortalidad, así como sobre otra información demográfica como series de fecundidad, nupcialidad, divorcios y migración. Asimismo se advierte su aplicación en otras áreas del conocimiento. Para este trabajo se utiliza el software RATS versión 7.

A partir de lo estudiado en la elaboración de esta tesis, se visualizan diversas líneas de investigación. Una de ellas es la desagregación de series de población por cohortes. Es decir, si se tienen las series parciales de la población por cohortes y la serie de población total, seria útil prever las series de las cohortes y utilizar los enfoques siguientes: a) considerar la información de que la suma de las predicciones de las cohortes es consistente con las predicciones del total y b) suponer que hay un factor que influye en todas las cohortes, lo que lleva a construir un modelo factorial, y generar predicciones combinando datos de cada serie y del total (Análisis factorial dinámico). Además, seria oportuno analizar la relación entre esos dos enfoques.

Por otro lado, se podría relacionar la necesidad de desagregar y suavizar simultáneamente información demográfica, así se podría pensar en una *desagregación suavizada*, donde el analista decidiera que nivel de suavidad deseado para propiciar la comparabilidad con otras tendencias de mortalidad u otro indicador demográfico. Con ello sería idóneo proponer índices de suavidad, deducir relaciones teóricas y estudiar sus propiedades.

Se considera que se tiene un amplio camino por recorrer en cuanto a los pronósticos restringidos sobre eventos demográficos que presenten volatilidad estocástica y que se pueden tratar a través de modelos de la familia ARCH, tanto en el caso univariante como en el multivariante. Dentro de dichos fenómenos se tienen identificados algunos que pudieran ser los siguientes: comportamientos especiales de migraciones, muertes en accidentes automovilísticos, la morbilidad derivada de la ocurrencia y expansión de epidemias o pandemias, el nivel de población económicamente activa en diversas áreas geográficas captadas a través de encuestas periódicas y en contextos económicos poco estables.

Otra línea de investigación, es la referente a combinación de información a partir de leyes teóricas de mortalidad (modelos paramétricos) y estructuras generales de mortalidad u otro fenómeno demográfico, tanto de países o regiones desarrolladas como en vías de desarrollo, donde al analista pueda otorgar determinado nivel de credibilidad a alguna o varias de las fuentes de información. Así podría ser interesante generalizar al manejo de fuentes y captar toda la dinámica histórica del fenómeno en estudio. Para este propósito podría ser apropiado el uso de optimización no lineal, la definición y uso de funciones de pérdida, así como tener presente la necesidad del desarrollo de habilidades para la elaboración de programas de cómputo para realizar cálculos que se vayan requiriendo.

Respecto a la propuesta del tercer trabajo, la metodología podría generalizarse al caso bidimensional, donde se prevé que, como ha ocurrido en el expuesto en la tesis, pudiera haber resultados teóricos interesantes en los que se relacionen los distintos parámetros de suavizamiento y donde sería pertinente poder aplicar la técnica para generar estimaciones de superficies mortalidad, restringidas a la experiencia y valoraciones que considere apropiadas el analista, con el propósito de graduar información y propiciar la comparabilidad. En términos prácticos, podría surgir la inquietud o requerimiento de aplicar la metodología por trozos sobre las series de mortalidad dentro del rango de edades, tanto para el caso unidimensional como en el bidimensional. Esta se podría presentar a partir de que el analista desee mucha mayor cercanía con una estructura demográfica en determinado rango y mantener el resto, por ejemplo, de manera equilibrada entre distintas fuentes de información.

En resumen, se prevén diversas líneas de investigación futuras y muy probablemente al desarrollar alguna de ellas, quedará constancia de que se está en una frontera conceptual muy rica por explorar en lo subsecuente entre el Análisis Demográfico y las series de tiempo.

Chapter 1. Introduction

The development of demographic analysis depends on the use of statistical methods which are derived from the needs of different scientifics fields. In particular time series has been a very important tool in the development of demographic analysis. Research papers that linked demography and time series are focusing on modeling, analyze and forecast demographic phenomena. On the *fertility* topic we can find, among others: Land and Cantor (1983); Carter and Lee (1986); De Beer (1989); Thompson et al. (1989); De Beer (1992); Lee (1993); Rogers et al. (1993); Bell (1997); Durban et al. (2001); McNown and Rajbhandary (2003); McNown and Ridao-Cano (2005) and Jeon and Shields (2008).

Some works related to *mortality* are, for instance, McNown and Rogers (1989a, 1989b and 1992); Laporte and Fergusonb (2003); Alonso (2008); Goldstein (2009). Others, linked with *migration* are: Brücker et al. (2003) and Cornwell (2009). Also, both initial and current works have proposed forecastig population, such as: De Beer (1985); Lee (1992); Lee and Tuljapurkar (1994); Keilman et al. (2002); Girosi and King (2004); Tuljapurkar et al. (2004); Hyndman and Booth (2008); Alonso et al. (2009) and Okita et al. (2009).

The importance of solving demographic problems, from perspective of time series, lies in allowing the decision maker to act beyond their beliefs, and so he can provide an appropriate environment for the formulation of population policies. In this sense, the objective of this thesis is to cover various situations where it is necessary to combine or smooth information contained in multiple kinds of demographic time series. The thesis consists of three main chapters that consider different issues that can be emerge in the demographic and the actuarial fields.

The Chapter 2 shows some applications of time series methods aimed to solve two typical problems that arise when analyzing demographic data in developing countries: (1) lack of existence of the annual series of population or their annual growth, and (2) inappropriate strategies for defining the goals of population growth in official population programs (supposedly based on its own historical record). These problems are seen as situations that require a combination of time series data on human populations. First, it is suggested the use of temporal disaggregation techniques to combine the decennial census data with annual information coming from vital statistics to estimate annual growth rates of the population. Second, multiple restricted forecasting technique is applied for combining multiple official goals of future rates of population growth with the disaggregated time series. Then, a mechanism is proposed for assessing the compatibility of population objectives with annual data. Then when the above procedures are applied to data from the Metropolitan Zone of Mexico City, divided by concentric rings, it is concluded that the goals established in the official program are not empirically feasible. Therefore, we infer future population growth rates that are consistent with the official targets and with the historical demographic behavior. We conclude that the programs of population growth must be based on this type of analysis in order to consider the empirical evidence.

In the Chapter 3, we present a method for choosing the smoothing constant to estimate trends of mortality rates with penalized splines (P-splines) in two dimensions, allowing the user to set a desired percentage of smoothness fixed beforehand, in both years and ages. The practical usefulness of this methodology is to allow comparability of mortality trends with equal percentages of smoothness. This procedure generalizes the method for choosing the smoothing parameter that produces univariate time series trends with smoothness set by the user, which arises from an index of smoothness. A theoretical result is provided to relate the smoothness index for both the one-dimensional and the two-dimensional cases. Some considerations related to numerical aspects and illustrative examples are presented in both cases.

In the Chapter 4, a non-parametric method is proposed to estimate trends in mortality rates, that combines the goodness of fit and smoothness of a non-parametric approach, with information from a given structure of mortality. In this way, the user is able to control both the smoothness and the structure of the estimated mortality. The main objective of this proposal is to be able to compare mortality trends with equal percentages of smoothness and pre-established structure. Two perspectives are emphasized in the proposed methodology: first, to compromise fit with desired smoothness, and on the other, to combine two sources of information, where the analyst can decide which of those two sources deserves more credibility. The usefulness of the method is illustrated through empirical examples that make use of various indicators of mortality.

The three chapters contain their specific conclusions, sections and appendices if necessary. They show charts, graphs and figures intended for a clear exposition of the issues under study. Within each work, a consecutive order of the sections and formulas is employed, and it should be stressed that each work is independent of the others. In the last section of this thesis, called “Conclusions and further research”, the main findings are emphasized and some lines of research are identified. It should be clear that the interaction between statistics and demographics is the key argument exploited here. Its importance lies in the possible use of statistical reasoning and the corresponding methods for decision making through the description and analysis of demographic data.

Chapter 2. Temporal disaggregation and restricted forecasting of multiple population time series

2.1 Introduction

Unavailability of annual population growth rates represents a problem for policy and decision makers, particularly in developing countries. This problem occurs in México in spite of the fact that census data are generated regularly every 10 years and that annual vital statistics of births and deaths are also available. Another problem is that inappropriate targets of population growth rates are usually proposed in the official programs for political reasons. Demographers typically apply easy-to-use, but suboptimal, tools to solve those problems. Besides, there is no unique solution to those problems due to the subjectivity involved in its application. For instance, a demographer would solve the previous problems by interpolating the census data to obtain annual data and then he/she would use personal beliefs to describe the patterns of fertility, mortality and migration in order to build scenarios of the future population growth. It should be clear that in such a case, it is no possible to associate a confidence level or credibility to the scenarios. This is in contrast with our proposal, because we suggest solving those problems from a statistical point of view and using optimality criteria. Another point worth emphasizing is that demographers tend to rely on univariate procedures, while our proposal consists of multivariate techniques.

Our proposal goes as follows, firstly we use a disaggregation technique to estimate time series of population growth, based primarily on census data and demographic information in the form of vital statistics; secondly, we employ a multiple restricted forecasting technique, with its compatibility testing companion, to analyze the official goals for population growth proposed by the Government. Thus, in order to estimate unavailable annual population data of the Metropolitan Zone of Mexico City (MZMC), we combine decennial census data with annual vital statistics using temporal disaggregation. The combination involves multiple time series data, since we consider that the MZMC is composed by the Central City and three concentric rings, as shown in Figure 2.1. The geographic units (delegations and municipalities) that compose these rings are available in Appendix 2.5.4. On the other hand, to evaluate the feasibility of the official goals for the population growth rate of each ring, we combine the targets with the annual disaggregated series. Thus, we generate multiple restricted forecasts with a Vector Auto-Regressive (VAR) model and carry out compatibility testing.

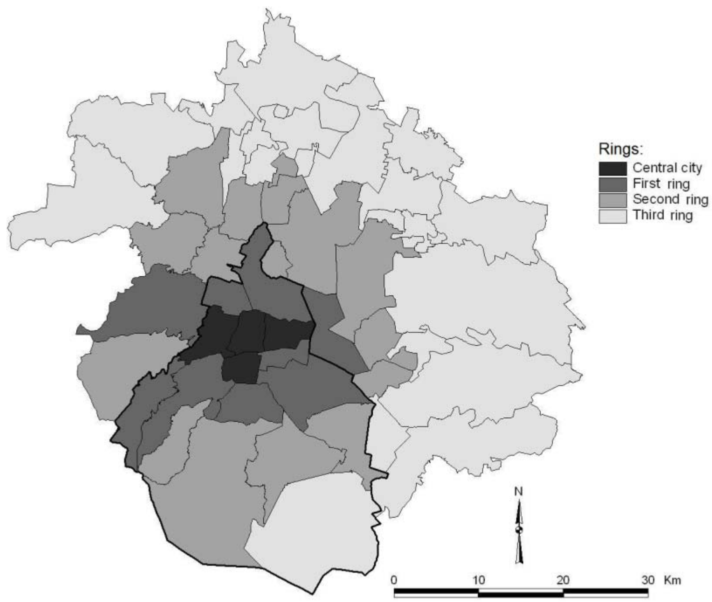


Figure 2.1. MZMC and its composition in concentric rings.

In Mexico, demographic data can be obtained from several sources of information, two of the most important are: (1) censuses carried out every 10 years (the most recent in year 2000) by the National Institute of Statistics and Geography (INEGI, Distrito Federal 1940-2000; INEGI, Estado de México 1940-2000), and (2) annual data on vital statistics given by births and deaths from 1940 up to 2000, for the Federal District (DF) and the State of Mexico (SM). These data can be obtained from the Secretariat of Health (SS, Distrito Federal 1993; SS, Estado de Mexico 1993), since 1940 up to 1993 and from INEGI (INEGI, Estadísticas vitales: Distrito Federal, Estado de México, 1994-2000), for years 1994 through 2000.

We propose to disaggregate low frequency demographic time series data on cumulative Population Growth Rates (PGR), available every decade, with the aid of auxiliary data observed with high frequency (annual vital statistics). Then, the resulting annual estimates will follow the annual pattern provided by the auxiliary data and satisfy the restrictions imposed by the census data. We apply a temporal disaggregation procedure to the census population series for each and every ring, including the Central City, and the resulting estimates will be reasonable in demographic terms, since the population of the rings will add up to the total population for the MZMC. The disaggregation technique that we will use is that proposed by Guerrero and Nieto (1999). Then, we shall employ multiple restricted forecasting, with its corresponding compatibility testing procedure, to evaluate the demographic targets established for the population growth rates of each ring. These targets appear in the population program for the DF, (Gobierno del DF y Consejo de Población del Distrito Federal, 1997).

. Some substantial results obtained in this work are the following. When using the temporal disaggregation technique, we obtained annual series estimates of cumulative PGR that behave as expected, according to the demographic logic. Besides, adequacy of the estimates for all rings was validated empirically by comparing them against data coming from an interdecade population counting. Then, when applying multiple restricted forecasting with the official targets as restrictions, we observed some incompatibility with the demographic dynamics and concluded that the proposed targets are not feasible. As a result, we proposed some other targets that became statistically compatible with the historical behavior (to reach this conclusion we performed a test at the 5% significance level). In particular for the case of Mexico, we did not find any trace of a previous work that focus on the demographic problem we deal with, neither with an approach similar to ours, nor with other approaches, to disaggregate series or to evaluate targets.

The rest of this chapter is organized as follows. In Section 2 we present the temporal disaggregation technique to be used and describe the procedure for multiple restricted forecasting, with its companion compatibility testing (for estimated processes). In addition, we show how to incorporate measurement error variability for variables measured with error (in our case, obtained by temporally disaggregating the census data). Section 3 illustrates the application of the aforementioned techniques to the four rings included in the MZMC. First, with temporal disaggregation we obtain estimated annual series of cumulative PGR for each ring and years 1940-2000. The second application provides us with multiple restricted forecasts for the concentric rings and allows us to analyze their respective compatibilities with the official targets. We make some comments about these targets and deduce feasible goals for the future PGR. In Section 4 we conclude with some final comments. The Appendixes show how we: (i) corrected the vital statistics series for outlying observations, (ii) generated the preliminary series required by the disaggregation procedure, and (iii) incorporated the measurement error in the restricted forecasting procedure, for a proper combination of the goals with the annual estimated series.

2.2 Methodology

2.2.1 Temporal disaggregation of multiple time series

Several proposals aimed at solving the temporal disaggregation problem of multiple time series are generalizations of univariate disaggregation procedures. The limit of those methods is that they assume specific structures for the random error involved: white noise (Rossi, 1982; Di Fonzo, 1990), random walk (Di Fonzo, 1994), or multivariate $AR(1)$ (Pavía, 2000). Therefore, they can be considered as general devices that are usually applied without taking into account the particular features of the data under study. As such, they are easy to apply than data-specific procedures, but their appropriateness cannot be judged empirically. Rather than assuming a specific structure *a priori*, we follow Guerrero and Nieto's suggestion (1999), of deducing the structure from the observed data and assume only that a VAR model of order p is appropriate to capture the dynamics of the random error. So, the main advantage of this approach is its objectivity, since it is fully supported and suggested by the data. Moreover, it is derived from theoretical results and produces a statistically optimal estimate of the disaggregated multiple time series. We consider these to be key elements and they should be underlined because the proposed approach will be employed for the first time (as far as we know) to demographic information. Besides, it is important to note that the vital statistics available correspond to the DF and the SM, not to the rings, and this fact precluded the use of the alternative disaggregation procedures.

It should be noticed that rather than disaggregating the multiple population time series, we could have used a different approach such as the Mixed Data Sampling (MIDAS) regressions as in Clements and Galvão (2008) and Ghysels et al. (2006). With such an approach we could merge data with different frequencies of observation (say decennial and annual) into a single regression equation to produce efficient forecasts of the higher frequency series. With this approach we would not require to disaggregate the series and could proceed directly to generate forecasts, but then we would have required to extend these ideas to the multiple equation case and derive the corresponding restricted forecasting formulas for this situation. Thus, it would be interesting to employ the MIDAS approach in future works and analyze its eventual improvement of the multiple unrestricted and restricted forecasts,

Let us first define $\boldsymbol{Z}_{Dt} = (Z_{it},...,Z_{kt})'$ as the k-dimensional column vector of non observable variables at time t, for $t = 1,...,mn$ where n is the number of complete periods and m is the intraperiod frequency (*m=10* years in a decade), while $\boldsymbol{Z}_D = (\boldsymbol{Z'}_{D1},...,\boldsymbol{Z'}_{Dmn})'$ is a stacked vector that contains the vectors $\boldsymbol{Z}_{Dt}$. Besides, $\boldsymbol{W}_{Dt}$ and $\boldsymbol{W}_D$ are defined as vectors of preliminary data corresponding to $\boldsymbol{Z}_{Dt}$ and $\boldsymbol{Z}_D$, respectively. We want to estimate $\boldsymbol{Z}_D$ on the basis of $\boldsymbol{W}_D$ and the identity

$$\boldsymbol{Y}_D = C_D \boldsymbol{Z}_D \tag{1}$$

where $\boldsymbol{Y}_D$ is a *kn*-dimensional vector that contains the aggregated data of $\boldsymbol{Z}_D$ and C_D is a known $kn \times kmn$ constant matrix. The following result was established in Guerrero and Nieto (1999).

Proposition. The Best Linear Unbiased Estimator of $\boldsymbol{Z}_D$, given $\boldsymbol{W}_D$ and $\boldsymbol{Y}_D$ is

$$\hat{\boldsymbol{Z}}_D = \boldsymbol{W}_D + A_D(\boldsymbol{Y}_D - C_D \boldsymbol{W}_D) \tag{2}$$

with

$$Cov(\hat{\boldsymbol{Z}}_D - \boldsymbol{Z}_D \mid \boldsymbol{W}_D) = (I_{knm} - A_D C_D)\Pi^{-1}(P \otimes \Sigma_a)\Pi^{-1}{}' \tag{3}$$

in which

$$A_D = \Pi^{-1}(P \otimes \Sigma_a)\Pi^{-1}{}' C_D{}' \left[C_D \Pi^{-1}(P \otimes \Sigma_a)\Pi^{-1} C_D{}' \right]^{+} \tag{4}$$

where the superscript + denotes Moore-Penrose inverse. The $kmn \times kmn$ matrix Π is built from the matrix coefficients $\pi_1,...,\pi_p$ of the polynomial involved by the VAR model, as follows

$$\Pi = \begin{bmatrix} I_k & 0 & 0 & 0 & 0 \\ -\pi_1 & I_k & 0 & 0 & 0 \\ & \vdots & \ddots & & \vdots \\ -\pi_p & -\pi_{p-1} & -\pi_{p-2} & 0 & 0 \\ 0 & -\pi_p & -\pi_{p-1} & 0 & 0 \\ & \vdots & \ddots & & \vdots \\ 0 & 0 & 0 & -\pi_1 & I_k \end{bmatrix}.$$

Moreover, P is an $mn \times mn$ positive definite matrix derived from the data and Σ_a is the error variance-covariance matrix of the VAR model. We refer the reader to the original paper Guerrero and Nieto (1999) for details on these definitions and the method itself. The operational procedure derived from this proposition consists of two stages. In the first stage we obtain a preliminary disaggregated series $\{\boldsymbol{W}_D\}$ on the basis of the theory underlying the phenomenon under study and fit a VAR model with deterministic terms to that series. From such a model and the expressions in Proposition 1 (with $P = I$) we obtain a tentatively estimated series $\{\bar{\boldsymbol{Z}}_D\}$ and test for whiteness of the series produced by $\hat{\Pi}(\bar{\boldsymbol{Z}}_D - \boldsymbol{W}_D)$. If this series behaves as white noise we conclude that the tentative series is statistically supported and call it the final disaggregated series. Otherwise, we go to the second stage where we look for a VAR representation of the differences in order to obtain an estimate of the matrix P and derive the final estimate $\hat{\boldsymbol{Z}}_D$, using again Proposition 1.

2.2.2 Multiple unrestricted forecasts

Let $\boldsymbol{Z}_t = (Z_{it}, \ldots, Z_{kt})'$ be a vector of k variables observed at time t, for $t = 1, \ldots, N$. In our case, the multiple time series $\{\boldsymbol{Z}_t\}$ comes from an application of the disaggregation procedure and admits the VAR(p) representation

$$\Pi(B)\boldsymbol{Z}_t = \Lambda \boldsymbol{D}_t + \boldsymbol{a}_t \tag{5}$$

where $\Pi(B)$ is a matrix polynomial of order p < ∝ in the backshift operator B such that $BX_t = X_{t-1}$ for every variable X and subindex t. $\boldsymbol{D}_t$ is a vector containing the deterministic variables (usually a constant and a linear trend), Λ is a matrix of coefficients that capture the deterministic effects and $\{\boldsymbol{a}_t\}$ is a k-dimensional Gaussian zero-mean white noise process with positive definite covariance matrix $E(\boldsymbol{a}_t \boldsymbol{a}'_t) = \Sigma_a$.

Further, let $\boldsymbol{Z} = (\boldsymbol{Z}'_1, ..., \boldsymbol{Z}'_N)'$ be the vector of known data and let $\boldsymbol{Z}_F = (\boldsymbol{Z}'_{N+1}, ..., \boldsymbol{Z}'_{N+H})'$ be the vector of future values, with $H \geq 1$ the forecast horizon. The optimal linear forecast of $\boldsymbol{Z}_{N+h}$, in minimum Mean Square Error (MSE) sense, is given for $h = 1,...,H$, by

$$E(\boldsymbol{Z}_{N+h} | \boldsymbol{Z}) = \Lambda \boldsymbol{D}_t + \Pi_1 E(\boldsymbol{Z}_{N+h-1} | \boldsymbol{Z}) + ... + \Pi_p E(\boldsymbol{Z}_{N+h-p} | \boldsymbol{Z}) \tag{6}$$

with $E(\boldsymbol{Z}_{N+h} | \boldsymbol{Z}) = \boldsymbol{Z}_{N+h}$ for $h \leq 0$. The corresponding forecast errors, are given by

$$\boldsymbol{Z}_F - E(\boldsymbol{Z}_F | \boldsymbol{Z}) = \boldsymbol{\Psi} \boldsymbol{a}_F \tag{7}$$

where $\boldsymbol{a}_F = (\boldsymbol{a}'_{N+1}, ..., \boldsymbol{a}'_{N+H})' \sim N(\boldsymbol{0}, I_H \otimes \Sigma_a)$ and $\boldsymbol{\Psi}$ is the $kH \times kH$ lower triangular matrix with the identity I_k in its main diagonal, Ψ_1 in its first subdiagonal, Ψ_2 in its second subdiagonal and so on. Where the Ψ matrices are obtained recursively from the following expressions

$$\Psi_0 = I_k, \quad \Psi_j = \Pi_j + \Pi_{j-1}\Psi_1 + \Pi_{j-2}\Psi_2 + ... + \Pi_1\Psi_{j-1} \text{ for } j = 1, ..., H-1, \tag{8}$$

with $\Pi_j = 0$ if $j > p$ or $j < 0$, see Wei (1990). Thus, the multiple unrestricted forecasts are conditionally unbiased and their MSE matrix is given by

$$MSE = \boldsymbol{\Psi}(I_H \otimes \Sigma_a)\boldsymbol{\Psi}'. \tag{9}$$

2.2.3 Multiple restricted forecasts

We now consider that some additional information is available in the form of a vector $\boldsymbol{Y} = (Y_1, ..., Y_M)'$ that imposes $M \geq 0$ independent linear restrictions on the future values of the vector $\boldsymbol{Z}$. These restrictions come from an external source to the time series model and are related to $\boldsymbol{Z}_F$ by means of

$$\boldsymbol{Y} = C\boldsymbol{Z}_F + \boldsymbol{u} \tag{10}$$

where $\boldsymbol{u} \sim N(\boldsymbol{0}_M, \Sigma_u)$. In our case, the restrictions are targets on the population rate of growth and in order to test for their compatibility with the unrestricted forecasts, we assume they are certain, so that $\Sigma_u = 0$. Besides, C is an $M \times kH$ matrix of rank M given by C = [$\boldsymbol{c}_1$ … $\boldsymbol{c}_M$]' where $\boldsymbol{c}_m = (c_{m,1}, \ldots, c_{m,kH})$ for $m = 1, \ldots, M$.

Using (7) and (10) Pankratz (1989), showed that the optimal restricted forecast of $\boldsymbol{Z}_F$ is

$$\boldsymbol{Z}^R_{F,H} = E(\boldsymbol{Z}_F | \boldsymbol{Z}) + A[\boldsymbol{Y} - CE(\boldsymbol{Z}_F | \boldsymbol{Z})] \tag{11}$$

with

$$MSE(\boldsymbol{Z}^R_{F,H}) = (I_H - AC)\Sigma_{E(Z_F|Z)} \text{ and } A = \Sigma_{E(Z_F|Z)} C' \Omega^{-1} \tag{12}$$

where

$$\Sigma_{E(\boldsymbol{Z}_F|\boldsymbol{Z})} = \boldsymbol{\Psi}(I_H \otimes \Sigma_a)\boldsymbol{\Psi}' \text{ and } \Omega = C\boldsymbol{\Psi}(I_H \otimes \Sigma_a)\boldsymbol{\Psi}' C'. \tag{13}$$

Expressions (11)-(13) can be obtained also by applying Theorem 1 of Nieto and Guerrero (1995) without the Normality assumption required by Pankratz´s result.

2.2.4 Compatibility Testing

Combining information should be judged from an empirical point of view, because the restrictions imposed to the series by the population goals may contradict the observed behavior of the series. To this end we use the following statistic proposed by Guerrero and Peña (2003, 2000),

$$K = \boldsymbol{d}' \Omega^{-1} \boldsymbol{d} \sim \chi^2_M \tag{14}$$

where $\boldsymbol{d} = \boldsymbol{Y} - CE(\boldsymbol{Z_F} | \boldsymbol{Z})$. Then, $\boldsymbol{Y} - CE(\boldsymbol{Z_F} | \boldsymbol{Z})$ lies in the compatibility region if the calculated statistic K_{calc} is not greater than $\chi^2_M(\alpha)$, the (1-α)-th quantile of a Chi-square distribution with M degrees of freedom, and we declare $\boldsymbol{Y}$ incompatible with $CE(\boldsymbol{Z_F} | \boldsymbol{Z})$ at the 100α% significance level if K_{calc} > $\chi^2_M(\alpha)$. We can also use partial compatibility test statistics, denoted as K_{par}, to evaluate the compatibility of specific restrictions with unrestricted forecasts.

2.2.5 VAR forecasting and compatibility testing with estimated processes

In a VAR model with estimated parameters the forecasts are conditionally unbiased and asymptotically valid (Dufour, 1985). Also, it can be shown that the vector of optimum restricted forecasts with an estimated process is given by

$$\hat{\boldsymbol{Z}}^R_{F,H} = E(\hat{\boldsymbol{Z}}_F|\boldsymbol{Z}) + \hat{A}\left[\boldsymbol{Y} - CE(\hat{\boldsymbol{Z}}_F|\boldsymbol{Z})\right] \tag{15}$$

where

$$\hat{A} = \hat{\Sigma}_{E(\hat{\boldsymbol{Z}}_F|\boldsymbol{Z})}C'\hat{\Omega}^{-1}, \quad \hat{\Omega} = C\hat{\Sigma}_{E(\hat{\boldsymbol{Z}}_F|\boldsymbol{Z})}C' \tag{16}$$

and

$$\hat{\Sigma}_{E(\hat{\boldsymbol{Z}}_F|\boldsymbol{Z})} \approx \hat{\Sigma}_{E(\boldsymbol{Z}_F|\boldsymbol{Z})} + N^{-1}\hat{\Sigma}_a. \tag{17}$$

Moreover, its estimated MSE matrix is given by

$$M\hat{S}E(\hat{\boldsymbol{Z}}^R_{F,H}) = (I - \hat{A}C)\hat{\Sigma}_{E(\hat{\boldsymbol{Z}}_F|\boldsymbol{Z})}. \tag{18}$$

Compatibility testing should also be modified for estimated processes. Gomez and Guerrero (2006), showed that the appropriate test statistic is given by

$$\overline{K} = \hat{\boldsymbol{d}}'\hat{\Omega}^{-1}\hat{\boldsymbol{d}}/M \sim F_{M,T-Mp-1} \tag{19}$$

where $\hat{\boldsymbol{d}} = \boldsymbol{Y} - CE(\hat{\boldsymbol{Z}}_F|\boldsymbol{Z})$. So that, $\boldsymbol{Y}$ is not in the compatibility region at the α significance level if $\overline{K}_{calc} \geq F_{M,T-Mp-1}(\alpha)$ with $F_{M,T-Mp-1}(\alpha)$ being the (1-α)-th quantile of the $F_{M,T-Mp-1}$ distribution. This statistic will be used below for examining compatibility between official targets and unrestricted forecasts. Similarly, we will apply partial compatibility test statistics, $\overline{K}_{par}$, to evaluate the compatibility between specific restrictions and unrestricted forecasts.

2.2.6 Incorporating measurement error variability

From now on we denote Central City population by ccp_t, First ring population by frp_t, Second ring population by srp_t, Third ring population by trp_t, MZMC population by $mzmcp_t$, DF population by dfp_t and SM population by smp_t. In this application, it is very important to note that the multiple VAR forecasts are not obtained from actual observations of the variables of interest, but from estimated data (hence, measured with error) derived as an application of the disaggregation technique. This is an important point that must be emphasized because VAR forecasts are generally produced from observed time series, which is not the case here. In fact, the VAR model is used to forecast an unobserved disaggregated multiple time series which came out from an unbiased estimation procedure. Hence, the estimated series will be assumed to be equal to the true, but unobserved, time series plus an error term, that we call a measurement error. In Appendix 2.5.3 we show how to incorporate the measurement error variability into the restricted forecasting formula.

Thus, to take into account these measurement errors into the forecasts we define the 80×80 matrix of estimated measurement error variances

$$\hat{\Sigma}_{\varepsilon} = I_{20} \otimes diag\left(\overline{\sigma}^2(ccp), \overline{\sigma}^2(frp), \overline{\sigma}^2(srp), \overline{\sigma}^2(trp)\right), \tag{20}$$

where $\otimes$ denotes Kronecker product and every element in the diagonal matrix is an average of the respective elements $\hat{\sigma}_t^2(ccp)$, $\hat{\sigma}_t^2(frp)$, $\hat{\sigma}_t^2(srp)$ and $\hat{\sigma}_t^2(trp)$ for t = 1981, 1982,…, 2000. These estimated variances are taken from the diagonal of the covariance matrices produced by the disaggregation procedure. We considered the errors of the last two decades because the forecasts are required for a 20-year horizon, from 2001 to 2020. The matrix $\hat{\Sigma}_{\varepsilon}$ was added to equations (17) and (18) to include the effect of measurement errors. Without it we could get a false idea of the variability associated with the multiple restricted forecasts, the compatibility test would not be strictly valid, and the evaluation of official targets would lead to erroneous conclusions.

It is convenient to mention that Nieto (2007), has provided another approach to solve essentially the same problem considered here. His solution is shown to produce optimal forecast in the context of the so-called ex-ante prediction of unobservable multivariate time series. Thus, it would be interesting to apply his results in a future work that postulate a multivariate structural model.

2.3 Applications

2.3.1 Application 1: Temporal disaggregation

In this application, temporal disaggregation of the census data is equivalent to interpolate them by annual figures. We require first preliminary series for each concentric ring and to get them it was necessary to correct the annual births series for outlying observations (see Appendix 2.5.1). Then, we employed the algorithm in Appendix 2.5.2 to focus the problem from a demographic, rather than a statistical point of view, see for example Chow and Lin (1971). We did that to obtain a better subject matter interpretation of the resulting annual population series.

The computations were performed with the packages E-Views 5.1 (Quantitative Micro Software) and Matlab 7 (see Appendix 2.5.5-2.5.10). The data are available from the authors on request. Let $zccp_t$, $zfrp_t$, $zsrp_t$ and $ztrp_t$ be the non-observable variables at time $t = 1941, 1942, ..., 2000$ representing the cumulative PGR of the Central City and the rings. The number of complete periods is n = 6 (decades) and m = 10 is the number of annual observations in a decade. Let $\boldsymbol{zccp} = (zccp'_1, ..., zccp'_{mn})'$ be a stacked vector of the mn values of $zccp$. The vectors

$\boldsymbol{zfrp}$, $\boldsymbol{zsrp}$ and $\boldsymbol{ztrp}$ are defined similarly. Then, we define the vectors of preliminary series $\boldsymbol{wccp}$**,** $\boldsymbol{wfrp}$**,** $\boldsymbol{wsrp}$ and $\boldsymbol{wtrp}$ corresponding to $\boldsymbol{zccp}$**,** $\boldsymbol{zfrp}$, $\boldsymbol{zsrp}$ and $\boldsymbol{ztrp}$.

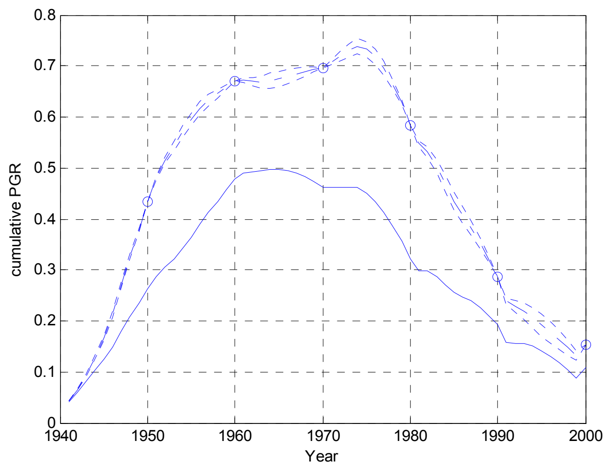
The temporal restrictions are specified by means of $\boldsymbol{Y}_D = (I_6 \otimes C_0)\boldsymbol{Z}_D$, where $C_0 = [\boldsymbol{0}_9\ 1]$ with $\boldsymbol{0}_9$ a 9-dimensional zero vector. The six elements of the vector $\boldsymbol{Y}_D$ are the cumulative PGR for the rings, coming from the census data, *i.e.* for the years ending in zero from 1950 up to 2000. No contemporaneous restrictions are used in this case, since they are considered implicitly by the temporal restrictions. Therefore, the multivariate application of this technique became a univariate application, and we applied the disaggregation procedure to each univariate time series (for each ring) separately.

In the first stage we built an autoregressive model to represent the behavior of the preliminary series for each ring. In the second stage we used another autoregressive model for the differences between the tentatively estimated series and the preliminary series. We present the estimation results in Table 2.1.

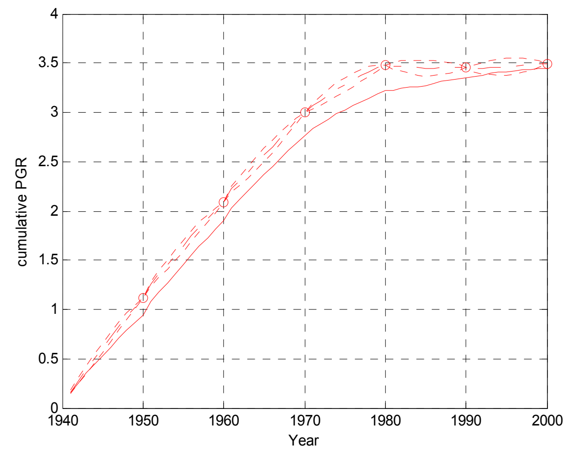
Table 2.1. Estimated autoregressive models used for univariate temporal disaggregation (t-statistics in parentheses)

Rings	First stage	Second stage
ccp_t	$\hat{wccp}_t = \underset{(29.67)}{1.881}\,\hat{wccp}_{t-1} - \underset{(-13.93)}{0.883}\,\hat{wccp}_{t-2}$ $\hat{\Sigma} = 7.42\times10^{-5}$	$\hat{\phi}_1 = \underset{(36.56)}{3.572},\ \hat{\phi}_2 = -\underset{(-17.11)}{4.820},\ \hat{\phi}_3 = \underset{(10.59)}{2.913},\ \hat{\phi}_4 = -\underset{(-7.29)}{0.666}$ $\hat{\Sigma} = 7.41\times10^{-8}$
frp_t	$\hat{wfrp}_t = \underset{(3.90)}{0.014}\,t - \underset{(-3.33)}{0.001}\,t^2 + \underset{(28.33)}{0.986}\,\hat{wfrp}_{t-1}$ $- \underset{(-10.31)}{0.110}\,\hat{wfrp}_{t-10}$ $\hat{\Sigma} = 1.16\times10^{-4}$	$\hat{\phi}_1 = \underset{(28.46)}{1.903},\ \hat{\phi}_2 = -\underset{(-12.90)}{0.916},\ \hat{\phi}_{10} = \underset{(7.60)}{0.686},\ \hat{\phi}_{11} = -\underset{(-8.52)}{1.339},$ $\hat{\phi}_{12} = \underset{(7.99)}{0.664}$ $\hat{\Sigma} = 1.33\times10^{-5}$
srp_t	$\hat{wsrp}_t = \underset{(4.32)}{0.001}\,t^2 - \underset{(-5.31)}{2.17\times10^{-5}}\,t^3 + \underset{(5.16)}{1.43\times10^{-7}}\,t^4$ $+\underset{(9.30)}{1.236}\,\hat{wsrp}_{t-1} - \underset{(-3.28)}{0.411}\,\hat{wsrp}_{t-2}$ $\hat{\Sigma} = 3.26\times10^{-4}$	$\hat{\phi}_1 = \underset{(20.92)}{2.375},\ \hat{\phi}_2 = -\underset{(-9.83)}{2.047},\ \hat{\phi}_3 = \underset{(6.10)}{0.650},\ \hat{\phi}_{10} = \underset{(12.51)}{0.851},$ $\hat{\phi}_{11} = -\underset{(-11.08)}{2.042},\ \hat{\phi}_{12} = \underset{(-11.079)}{1.777},\ \hat{\phi}_{13} = -\underset{(-5.42)}{0.572}$ $\hat{\Sigma} = 2.52\times10^{-5}$
trp_t	$\hat{wtrp}_t = \underset{(4.98)}{2.10\times10^{-4}}\,t^2\ - \underset{(-5.71)}{2.39\times10^{-8}}\,t^4 + \underset{(10.45)}{1.30}\,\hat{wtrp}_{t-1}$ $- \underset{(-4.25)}{0.466}\,\hat{wtrp}_{t-2}$ $\hat{\Sigma} = 7.49\times10^{-5}$	$\hat{\phi}_1 = \underset{(24.69)}{2.514},\ \hat{\phi}_2 = -\underset{(-10.97)}{2.176},\ \hat{\phi}_{13} = \underset{(6.49)}{0.659}$ $\hat{\Sigma} = 2.35\times10^{-5}$

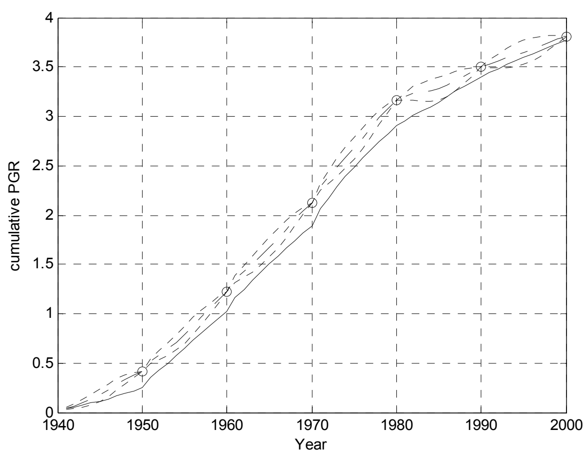
Standard errors for the disaggregated series were obtained as square roots of the elements in the diagonal of the estimated covariance matrix (3). Then, we obtained probability intervals (PI) from these estimates. These PI look as "bubbles" in Figure 2.2, because there is no uncertainty associated with the observed values for the census years.



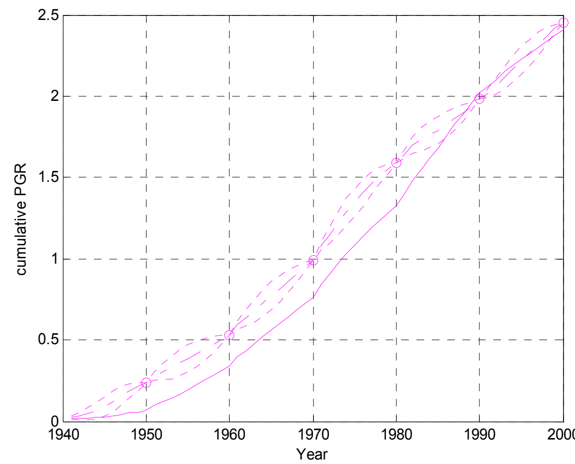
ccp_t



frp_t



srp_t



trp_t

Figure 2.2. Temporal disaggregation of — ccp_t, — frp_t, — srp_t, — trp_t. Solid lines denote preliminary series, dashed lines are disaggregate series with their 95% probability intervals and dots are census data.

The polynomials of order 4 in the models for the Second and Third rings look strange, but they were required to get a stationary behavior of their stochastic structures, since the method assumes that all kind of nonstationarities in the data can be taken into account by way of deterministic elements. Thus, all models used in the first and second stages have characteristic polynomials with roots outside the unit circle. Moreover, we could not reject the white noise hypotheses for the univariate residuals at the 5% significance level. As it was expected, the annual disaggregated series satisfy the temporal restrictions imposed by the observed census data. Some selected results appear in Table 2.2.

Table 2.2 Disaggregated series: Preliminary and Final, with standard errors (100×SE)

Year	Prelim.	Final	SE	Prelim.	Final	SE	Prelim.	Final	SE	Prelim.	Final	SE
		ccp_t			frp_t			srp_t			trp_t	
1941	0.042	0.043	0.001	0.156	0.163	0.031	0.040	0.042	0.018	0.019	0.022	0.017
1942	0.061	0.065	0.003	0.257	0.276	0.057	0.057	0.065	0.047	0.020	0.031	0.045
1943	0.080	0.092	0.005	0.352	0.388	0.078	0.075	0.094	0.079	0.021	0.045	0.076
1944	0.101	0.126	0.008	0.443	0.499	0.092	0.094	0.130	0.107	0.024	0.065	0.104
1945	0.126	0.168	0.011	0.532	0.610	0.098	0.117	0.176	0.125	0.029	0.092	0.123
1946	0.149	0.213	0.012	0.617	0.718	0.096	0.139	0.225	0.129	0.033	0.120	0.128
1947	0.179	0.269	0.012	0.705	0.828	0.085	0.167	0.282	0.117	0.044	0.155	0.117
1948	0.207	0.324	0.010	0.788	0.931	0.066	0.194	0.335	0.089	0.052	0.186	0.090
1949	0.235	0.379	0.006	0.867	1.027	0.038	0.219	0.381	0.049	0.058	0.212	0.050
1950	0.264	0.434	0.000	0.945	1.115	0.000	0.247	0.417	0.000	0.067	0.236	0.000
…	…	…	…	…	…	…	…	…	…	…	…	…
1991	0.157	0.239	0.012	3.365	3.444	0.070	3.440	3.528	0.080	2.059	2.016	0.055
1992	0.156	0.228	0.023	3.380	3.446	0.115	3.482	3.560	0.148	2.104	2.062	0.104
1993	0.156	0.219	0.033	3.395	3.451	0.145	3.524	3.591	0.200	2.149	2.114	0.142
1994	0.150	0.206	0.040	3.406	3.456	0.162	3.562	3.619	0.231	2.190	2.167	0.167
1995	0.141	0.192	0.043	3.415	3.462	0.168	3.598	3.648	0.242	2.228	2.219	0.175
1996	0.130	0.177	0.043	3.422	3.469	0.164	3.632	3.679	0.233	2.265	2.270	0.168
1997	0.117	0.163	0.039	3.429	3.476	0.147	3.666	3.712	0.203	2.302	2.320	0.145
1998	0.103	0.148	0.030	3.435	3.484	0.117	3.700	3.747	0.151	2.338	2.368	0.107
1999	0.088	0.132	0.017	3.442	3.489	0.072	3.733	3.781	0.082	2.374	2.413	0.058
2000	0.109	0.153	0.000	3.447	3.492	0.000	3.766	3.811	0.000	2.410	2.454	0.000

To validate the previous results empirically, we made use of the data obtained in an interdecade population counting carried out in Mexico in 1995. Table 2.3 shows the observed population figures obtained in that counting, see INEGI (1995). All the corresponding values for the rings fall within the 95% PI for the disaggregated values.

Table 2.3. Observed and disaggregated cumulative PGR values for 1995

Rings	Observed (interdecade counting)	Lower 95% limit	Estimated by disaggregation	Upper 95% limit
ccp_t	0.195	0.171	0.192	0.213
frp_t	3.485	3.279	3.462	3.545
srp_t	3.706	3.529	3.648	3.767
trp_t	2.255	2.132	2.219	2.305

2.3.2 Application 2. Evaluating population goals

To obtain multiple unrestricted forecasts, we first estimated a VAR model for the population series selecting its order by the Likelihood Ratio testing scheme with upper bound p = 5. The deterministic element in each equation of the VAR model was only a constant. The results are

$$H_0^1 : \pi_5 = 0 \quad vs.\ H_1^1 : \pi_5 \neq 0,\ \chi^2(16) = 6.69$$
$$H_0^2 : \pi_4 = 0 \quad vs.\ H_1^2 : \pi_4 \neq 0 \,|\, \pi_5 = 0,\ \chi^2(16) = 11.85$$
$$H_0^3 : \pi_3 = 0 \quad vs.\ H_1^3 : \pi_3 \neq 0 \,|\, \pi_5 = \pi_4 = 0,\ \chi^2(16) = 13.96$$
$$H_0^4 : \pi_4 = 0 \quad vs.\ H_1^4 : \pi_2 \neq 0 \,|\, \pi_5 = \pi_4 = \pi_3 = 0,\ \chi^2(16) = 64.17$$

so that p = 2 was deemed reasonably adequate. The estimated arrays $\hat{\pi}_1, \hat{\pi}_2, \hat{D}_t$ and $\hat{\Sigma}_Z$ are as follows (t-values in parentheses and – denotes a nonsignificant coefficient, at the 5% level)

$$\hat{\pi}_1 = \begin{bmatrix} \underset{(10.02)}{1.516} & - & - & - \\ - & \underset{(8.06)}{1.505} & - & - \\ - & - & \underset{(6.18)}{1.387} & - \\ - & - & - & \underset{(6.07)}{1.307} \end{bmatrix}, \hat{\pi}_2 = \begin{bmatrix} \underset{(-4.22)}{-0.619} & - & - & - \\ - & \underset{(-2.27)}{-0.451} & - & - \\ - & - & - & - \\ - & - & - & - \end{bmatrix}, \hat{D}_t = \begin{bmatrix} - \\ \underset{(1.98)}{0.039} \\ - \\ - \end{bmatrix}, \hat{\Sigma}_Z = 10^{-5} \begin{bmatrix} 6.64 & -0.13 & 1.28 & 2.63 \\ -0.13 & 16.2 & 16.9 & 5.67 \\ 1.28 & 16.9 & 34.2 & 13.0 \\ 2.63 & 5.67 & 13.0 & 8.59 \end{bmatrix}$$

The residuals produced individual Ljung-Box statistics that do not led us to reject the white noise hypotheses at the 5% significance level. That is,

$$\{zccp_{\mathrm{t}}\}: \; Q(10) = 16.047, \quad Q(20) = 30.005, \;\; Q(30) = 35.244,$$

$$\{zfrp_{\mathrm{t}}\}: \; Q(10) = 9.122, \quad Q(20) = 22.267, \;\; Q(30) = 29.001,$$

$$\{zsrp_{\mathrm{t}}\}: \; Q(10) = 12.157, \quad Q(20) = 16.757, \;\; Q(30) = 21.084,$$

$$\{ztrp_{\mathrm{t}}\}: \; Q(10) = 20.386^*, \; Q(20) = 30.079, \;\; Q(30) = 33.705$$

(* in this case, the individual Ljung-Box statistic does not reject the white noise hypothesis at the 1% significance level).

We also computed the multivariate portmanteau statistic $\hat{Q}_h = T^2 \sum_{j=1}^{h} (T-j)^{-1} tr(\hat{C}_j' \hat{C}_0^{-1} \hat{C}_j \hat{C}_0^{-1})$ where $\hat{C}_j = \sum_{t=j+1}^{T} \hat{a}_t \hat{a}'_{t-j} \Big/ T$ and $\hat{a}_t$ are the *k*-dimensional residuals of the estimated VAR(p) model, with $\hat{Q}_h \sim \chi^2\left(k^2(h-p)\right)$, where *k=4* is the number of variables, *p=2* is the lag order of the fitted model and *h* was chosen as 6 (for details on the use of this test, see Lütkepohl 2005). We obtained $\hat{Q}_6 = 43.466$ and compared this value to $\chi^2(64)_{0.95} = 83.68$ so that we could not reject the white noise hypothesis for the errors at the 5% level.

2.3.3 Unrestricted Forecasts

The 2005 Population interdecade census reported population figures for every geographic unit considered in the MZCM, see INEGI (2005). However, in 2009 INEGI made some adjustments to those figures and produced new estimated population figures. The official cumulative PGR, its forecast for each and every ring and its corresponding 95% PI are shown below (Table 2.4). There we see that all the official cumulative PGR figures fall within its probability interval.

Table 2.4. Officially estimated and forecasted cumulative PGR values for 2005

Rings	Estimated figure (interdecade counting)	Lower 95% limit	Unrestricted forecast	Upper 95% limit
ccp_t	0.147	0.034	0.104	0.175
frp_t	3.477	3.350	3.458	3.567
srp_t	3.866	3.856	4.008	4.160
trp_t	2.694	2.534	2.627	2.720

In Figure 2.3 we show the multiple unrestricted forecasts together with their probability bands and the official figures reported in 2009.

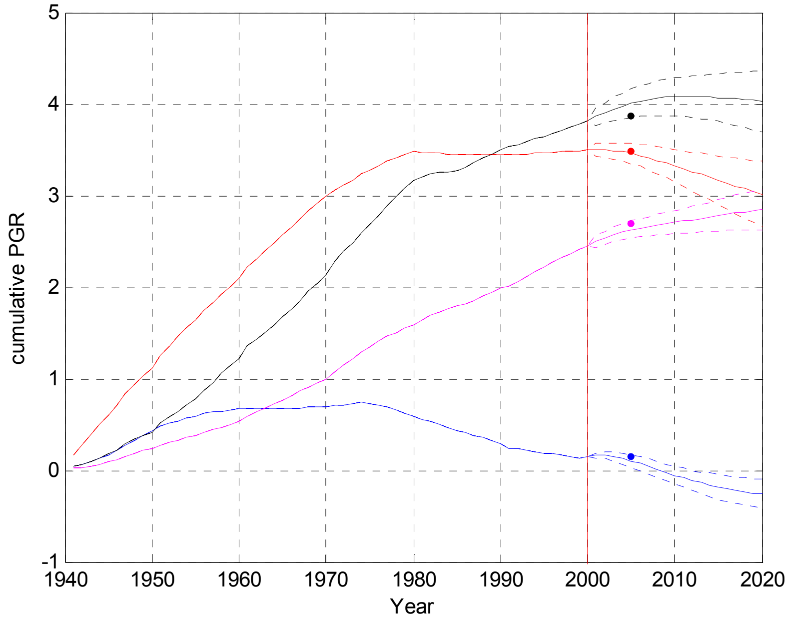


Figure 2.3. Unrestricted forecasts for 2001-2020 for — ccp_t, — frp_t, — srp_t, — trp_t with 95% probability intervals. Dots are official figures for ccp_t, frp_t, srp_t and trp_t.

2.3.4 Restricted forecasts and compatibility testing

In 1997, the DF Government presented a population program (see Gobierno del DF y Consejo de Población del Distrito Federal, 1997). Part B of that program included intended growth rates for the Central City and the rings. The specific demographic goals for population growth of the rings are: a) to reach a growth rate of 0.4% between 2006-2010 and 0.9% between 2010-2020 for the Central City; b) to reduce the growth rate to 0.3% between 2000-2003, increase it to 0.5% in 2006-2010 and reduce it to 0.3% between 2010-2020 for the First ring; c) to reduce the growth rate to 1.2% between 2000-2003, to 1.1% between 2003-2006, to 0.7% of 2006-2010 and to 0.5% in the following decade for the Second ring; d) to reduce the growth rate to 2.4% between 2000-2003, to 2.2% between 2003-2006, to 0.8% between 2006-2010 and to 0.7 in the following decade for the Third ring. We understood these values as goals to be reached at the end of every period and translated them into binding restrictions to be imposed on the forecasts of the cumulative PGR, as shown below (Table 2.5).

Table 2.5. Restricted forecasts for the concentric rings

Restricted forecast	Year			
	2003	2006	2010	2020
$\hat{zccp}_t$	without target	without target	$F\hat{zccp}_{2005} + 0.004^5$	$\hat{zccp}_{2010} + 0.009^{10}$
$\hat{zfrp}_t$	$zfrp_{2000} + 0.003^3$	without target	$F\hat{zfrp}_{2005} + 0.005^5$	$\hat{zfrp}_{2010} + 0.003^{10}$
$\hat{zsrp}_t$	$zsrp_{2000} + 0.012^3$	$\hat{zsrp}_{2003} + 0.011^3$	$\hat{zsrp}_{2006} + 0.007^4$	$\hat{zsrp}_{2010} + 0.005^{10}$
$\hat{ztrp}_t$	$ztrp_{2000} + 0.024^3$	$\hat{ztrp}_{2003} + 0.022^3$	$\hat{ztrp}_{2006} + 0.008^4$	$\hat{ztrp}_{2010} + 0.007^{10}$

Note: $zfrp_{2000}$, $zsrp_{2000}$, $ztrp_{2000}$ are observed 2000 census data. $F\hat{zccp}_{2005}$ and $F\hat{zfrp}_{2005}$ are unrestricted forecasts of ccp_t and frp_t produced by the VAR(2) model.

To take the previous restrictions into account, we define the ***Y*** vector as in (10) and the C matrix with the following structure

$$C = \begin{bmatrix} 0_{3\times 9} & I_3 & 0_{3\times 68} \\ 0_{2\times 22} & I_2 & 0_{2\times 56} \\ 0_{4\times 36} & I_4 & 0_{4\times 40} \\ \multicolumn{2}{c}{0_{4\times 76}} & I_4 \end{bmatrix}$$

where $0_{i\times j}$ are $i\times j$ zero matrices and I_i are i-dimensional identity matrices.

We carried out compatibility testing of these goals and obtained the value $K_{calc} = 3.45$ which is significant at the 5% level, as compared with an $F_{M,T-Mp-1}$ distribution with $M = 13$ and $T - Mp - 1 =$ 33 degrees of freedom. Therefore, the goals are jointly incompatible with the expected behavior of the multiple population series. However, the partial compatibility tests indicate that the goals $\hat{zccp}_{2010}$ and $\hat{zccp}_{2020}$ are may be considered compatible at the 0.07% and 0.05% significance level respectively.

Although the set of goals established in the population program are not jointly compatible, we shall elaborate on them and make a proposal on the population growth rates for the rings. The idea is to find a set of population targets that are compatible with the empirical evidence provided by the annual disaggregated series. Our proposal looks for population targets that produce a smooth population pattern, in agreement with the demographic logic, if no catastrophic or anomalous situation occurs. By so doing, we obtained the multiple compatibility test statistic $K_{calc} = 0.74$ with p-value 7.12%.

Since the unrestricted forecast for the Central City population has a clear decreasing trend, we suggest reaching a cumulative PGR of zero at the end of 2010 and fix a negative cumulative PGR of 0.18% at the end of 2020. For the First ring, all the targets of population growth were compatibles, but to get a smooth pattern of population we propose a cumulative PGR of 3.5% at the end of 2003, 3.38% in 2010 and 3% in 2020.

Our proposal for the cumulative PGR, based on the demographic dynamics presented by the series for the Second ring, is 3.9% at the end of 2003 and 3.98% in 2006, then it should go up to 4.06% in 2010 and 4.1% at the end of 2020. For the Third ring, our proposal is to modify only the first target at the end of 2003, that is, to reach a cumulative PGR of 2.55%, then reach 2.64% at the end of 2006, 2.68% in 2010 and 2.77% in 2020. In Table 2.6 we see that all the individual restrictions of our proposal are compatible with the disaggregated series at the 5% significance level, so that they are empirically supported.

Table 2.6. Compatibility testing for growth rates with our proposal

Restriction	K_{parc}	*M, T-Mp-1*	Significance
frp_{2003}	0.073	1, 57	0.788
srp_{2003}	0.550	1, 57	0.461
trp_{2003}	0.400	1, 57	0.530
srp_{2006}	0.368	1, 57	0.547
trp_{2006}	0.013	1, 57	0.910
ccp_{2010}	1.196	1, 57	0.279
frp_{2010}	0.374	1, 57	0.544
srp_{2010}	0.031	1, 57	0.862
trp_{2010}	0.197	1, 57	0.659
ccp_{2020}	1.052	1, 57	0.309
frp_{2020}	0.004	1, 57	0.951
srp_{2020}	0.190	1, 57	0.665
trp_{2020}	0.394	1, 57	0.533

Finally, in Figure 2.4 we can see the expected behavior of the population series for the Central City and the rings. The observed patterns are reasonably smooth for all the series except for the Central City population. We think this is a consequence of imposing constraints on that series that essentially tend to lower its cumulative PGR, so that the restricted forecasts have to bend the smooth curve in order to fulfill the constraints. In summary, we conclude that the goals proposed for the Central City should be different than those presented in the official population program, while those for the rings must be in general only slightly different.

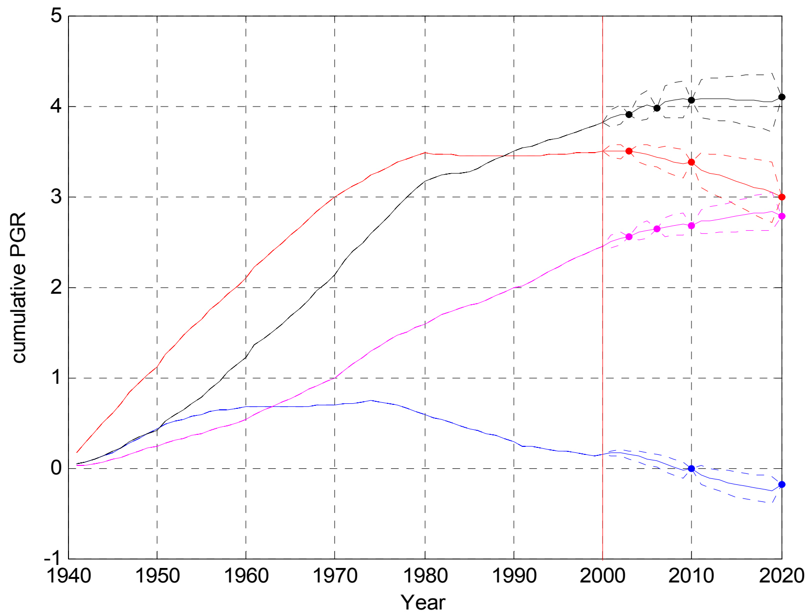


Figure 2.4. Restricted forecasts and 95% probability intervals for the rings with our proposal for — ccp_t, — frp_t, — srp_t, — trp_t,. Dots are the proposed goals.

2.4 Conclusions

We presented first an application of a temporal disaggregation technique to a demographic time series. The most time consuming part of such a technique involved the generation of an appropriate preliminary series from demographic considerations. We think it was worth doing it this way because the quality of the final results depends heavily on the quality of such a series. This task is much simpler to perform in economic contexts, because usually there are economic indexes that play the role of a basic auxiliary variable when obtaining a preliminary estimate of the unobserved series. In the case considered here, we were forced to perform a meticulous search for demographic data and events by geographic unit and year.

The application of restricted forecasting and compatibility testing to demographic data was carried out in order to evaluate the feasibility of the targets proposed in an official population program for the Metropolitan Zone of Mexico City. This analysis indicates that before suggesting demographic goals, it is necessary to evaluate their empirical feasibility in an objective way.

In general, it can be noticed that temporal disaggregation produced convincing results because we took great care to generate an adequate preliminary series. Also, it was possible to determine new goals consistent with the population dynamics. Of course, the methodological strategies presented here can be also used to solve similar problems with demographic information in other developing countries or in any other geographic zone where the need of combining demographic information arises. In fact, according with the results obtained in this paper, we could say that both temporal disaggregation and restricted forecasting are efficient statistical tools that serve to consolidate this type of data. Moreover, we are convinced that growth population programs could be made feasible and monitored afterwards with this kind of analysis.

This chapter provides evidence on the appropriateness and accessibility of specialized statistical techniques, that have been developed and traditionally employed for economic analysis, in the demographic field. We hope this work motivates specialists in these fields to identify new potential applications or possibilities of methodological developments, that will ultimately help practitioners to get more information from their data and support better decision making. It is worth stressing that the procedures applied here can be used with other kind of demographic data, such as those related with fertility, marriage, divorce and migration. By so doing, we could evaluate the feasibility of official population programs for the population determinants jointly, in different contexts and around the world.

2.5 Appendix

2.5.1 Correcting the annual birth series for outliers

We used annual vital statistics of births and deaths for years t = 1941, 1942, …, 2000 for the DF and the SM. The vital statistics in year t are: ddf_t deaths in the DF, bdf_t births in the DF, $ngdf_t = bdf_t - ddf_t$ natural growth in the DF, dsm_t deaths in the SM, bsm_t births in the SM and $ngsm_t = bsm_t - dsm_t$ natural growth in the SM. We corrected the annual $\{bsm_t\}$ series for outliers in years 1983, 1984 and 1985. This series was regressed on marriages nsm_t of the SM, a dummy variable $i77_t$ that accounts for a structural change in 1977 due to a birth control policy (valued zero from 1940 up to 1976 and 1 from 1977 onwards), three dummies for the aforementioned outliers i_{it} with i = 1,2,3 and lagged values of births bsm_{t-1} to account for its own inertia. Optimal corrected values were obtained as indicated in Gómez et al (1999), from the model expressed in logaritms and t = 1941, 1942, …, 2000,

$$log\ (bsm_t) = ß_1 i77_t + ß_2\, log\, (nsm_{t-1}) + \sum_{i=3}^{5} ß_i i_{it} + ß_6\, log\ (bsm_{t-1}) + a_t$$

where a_t is a random error from a zero-mean Gaussian white noise process. Since the dummy variable i_{3t} was not significant at the 5% level, it was omitted. The results of the final estimated model are (t-statistics in parentheses),

$$\hat{ß}_1 = \underset{(-2.13)}{-0.11},\ \hat{ß}_2 = \underset{(-3.18)}{0.21},\ \hat{ß}_3 = \underset{(-6.75)}{-0.68},\ \hat{ß}_4 = \underset{(-2.30)}{-0.24},\ \hat{ß}_6 = \underset{(-15.42)}{0.83}$$

with R^2 = 0.977 and $\hat{\sigma}_a = 0.098$. The Ljung-Box Q statistic for serial autocorrelation took on the followings values $Q(3) = 1.23, Q(5) = 2.98$ and $Q(10) = 13.17$, which did not provide evidence of inadequacy of the model.

2.5.2 Obtaining the preliminary series

The algorithm to get preliminary series for the rings from data on the DF and the SM goes as follows.

I. Calculate $ngdf_t$ and $ngsm_t$, for t = 1941, 1942, …, 2000.

II. Calculate dfp_t and smp_t as partial populations of the DF and the SM $dfp_t = dfp_{2000-n} + \sum_{j=2000\text{-}n+1}^{2000} ngdf_j$ and $smp_t = smp_{2000-n} + \sum_{j=2000\text{-}n+1}^{2000} ngsm_j$ with $n = 60$.

III. Consider the population proportions of the rings with respect to $dfp_{t*} + smp_{t*}$ for the census years $t*$ = 1940, 1950, …, 2000, represented by $rccp_{t*}$, $rfrp_{t*}$, $rsrp_{t*}$ and $rtrp_{t*}$. Calculate the proportions for the inter-census years *t,* given by $rccp_t$, $rfrp_t$, $rsrp_t$ and $rtrp_t$, assuming a linear behavior. The series of estimated proportions satisfy the relationship $rmzmcp_t = rccp_t + rfrp_t + rsrp_t + rtrp_t$ for t = 1941, 1942, …, 2000.

IV. Calculate the series of population proportions for the rings with $ccp_t^+ = rccp_t(dfp_t + smp_t)$, $frp_t^+ = rfrp_t(dfp_t + smp_t)$, $srp_t^+ = rsrp_t(dfp_t + smp_t)$ and $trp_t^+ = rtrp_t(dfp_t + smp_t)$ for t = 1941, 1942, …, 2000.

V. Calculate the differences attributable to migration for the rings: $migccp_{t*} = ccpc_{t*} \text{-} ccp_{t*}^+$, $migfrp_{t*} = frpc_{t*} \text{-} frp_{t*}^+$, $migsrp_{t*} = srpc_{t*} \text{-} srp_{t*}^+$ and $migtrp_{t*} = trpc_{t*} \text{-} trp_{t*}^+$ for $t*$= 1940, 1950, …, 2000, where $ccpc_{t*}$, $frpc_{t*}$, $srpc_{t*}$ and $trpc_{t*}$ are census populations at years $t*$. Suppose migration behaves uniformly in time and add one tenth of these differences to the annual estimates series obtained in the previous step.

VI. Calculate $ccp_t = ccp_t^+ + 0.1 * migccp_{t*}$, $frp_t = frp_t^+ + 0.1 * migfrp_{t*}$, $srp_t = srp_t^+ + 0.1 * migsrp_{t*}$ and $trp_t = trp_t^+ + 0.1 * migtrp_{t*}$ for t = 1941, 1942, …, 2000.

VII. Finally, estimate the cumulative population growth rate between 1940 and each year from 1941 to 2000, for every ring, as $r = \log_e(p_t / p_{1940})$.

2.5.3 Incorporating measurement error variability

Let $\{\boldsymbol{Z}^*_t\}$ and $\{\boldsymbol{Z}_t\}$ be series of observable and unobservable values, respectively, that admit stationary VAR representations of order 1 ≤ p < ∝ with all nonstationarities taken into account by the deterministic effects, so that their Wold's representations are $\boldsymbol{Z}^*_t = \boldsymbol{D}_t + \Psi(B)\boldsymbol{a}_t$ and $\boldsymbol{Z}_t = \boldsymbol{D}_t + \Psi(B)\boldsymbol{a}_t + \boldsymbol{\varepsilon}_t$ with $\boldsymbol{D}_t$ a vector of deterministic components that includes the constant term, $\boldsymbol{a}_t \sim N(\boldsymbol{0}, \Sigma_{\boldsymbol{a}})$ and $\boldsymbol{\varepsilon}_t$ is the measurement error which we assume is uncorrelated with the whole sequence $\{\boldsymbol{a}_t\}$. Then, we recall the notation in Section 2.2 to write

$$\boldsymbol{Z}^* = \boldsymbol{D} + \boldsymbol{\Psi}\boldsymbol{a}_F \text{ and } \boldsymbol{Z} = \boldsymbol{D} + \boldsymbol{\Psi}\boldsymbol{a}_F + \boldsymbol{\varepsilon}_F \tag{A1}$$

with $\boldsymbol{a}_F \sim N(\boldsymbol{0}, I_H \otimes \Sigma_{\boldsymbol{a}})$ and $\boldsymbol{\varepsilon}_F \sim N(\boldsymbol{0}, \Sigma_\varepsilon)$ and $E(\boldsymbol{a}_F \boldsymbol{\varepsilon}'_F) = 0$, where Σ_ε is given in equation (20) and $\boldsymbol{D}$ is a vector of deterministic elements. We then write $\boldsymbol{Z}_F = \boldsymbol{D} + \boldsymbol{\Psi}\delta_F$ with

$$\delta_F \sim N\left(\boldsymbol{0}, (I_H \otimes \Sigma_{\boldsymbol{a}}) + \boldsymbol{\Psi}^{-1}\Sigma_\varepsilon \boldsymbol{\Psi}^{-1\,\prime}\right) \tag{A2}$$

to obtain again expressions (11)-(13), with

$$A = \left[\boldsymbol{\Psi}(I_H \otimes \Sigma_{\boldsymbol{a}})\boldsymbol{\Psi}' + \Sigma_\varepsilon\right] C' \Omega^{-1} \tag{A3}$$

and

$$\Omega = C\left[\boldsymbol{\Psi}(I_H \otimes \Sigma_{\boldsymbol{a}})\boldsymbol{\Psi}' + \Sigma_\varepsilon\right] C'. \tag{A4}$$

2.5.4 Composition of each ring

To avoid confusions regarding the nomenclature of the geographical regions of study, the following annotations are made. The name of the country is Mexico (officially Mexican United States), the capital of the country it is Mexico City (that from now on it is denominated Central city) and the federative entity that it conforms part of MZMC it is State of Mexico (SM). The Central city belongs to the Federal District (DF), which is subdivided in 16 units denominated delegations. On the other hand, SM, according to the Census National Population and Housing of the 2000, was constituted by 123 units denominated municipalities, of which 33 are only inside MZMC, according to that adopted in this work. In summary, MZMC is a territory formed by the Central city, the remaining delegations of DF and 33 municipalities of SM.

The composition peculiar of each ring is the following one. Central city: delegations Benito Juárez, Cuauhtémoc, Miguel Hidalgo and Venustiano Carranza; First contour: delegations Azcapotzalco, Coyoacán, Cuajimalpa of Lives them, Gustavo A. Timber, Iztacalco, Iztapalapa and Alvaro Obregón, and municipalities Naucalpan of Juárez and Nezahualcóyotl; Second contour: delegations Magdalena Contreras, Tláhuac, Tlalpan and Xochimilco, and municipalities Atenco, Atizapán of Zaragoza, Coacalco of Berriozábal, Chimalhuacán, Ecatepec of Lives them, Huixquilucan, The Peace, Tlalnepantla of Baz, Tultitlán and Cuautitlán Izcalli; and Third contour: delegation High Cornfield, and municipalities Acolman, Cuautitlán, Chalco, Chiautla, Chicoloapan, Chiconcuac, Ixtapaluca, Jaltenco, Melchor Ocampo, Nextlalpan, Nicolás Romero, Papalotla, Tecámac, Teoloyucán, Teotihuacán, Tepetlaoxtoc, Tepotzotlán, Texcoco, Tezoyuca, Tultepec and Zumpango.

2.5.5 Matlab routine: ccp disagregation

```
%PRIMERA ETAPA
%ESTA DESAGREGACIÓN UNIVARIADA SIRVE PARA GENERAR LA SERIE UNIVARIADA DE CCP RESTRINGIDA
%TEMPORALMENTE
```

```
clear all
format long g

n=6;
m=10;
k1=1; % número de series
I21=eye(k1*m*n);%60x60
I3=eye(n);%6x6
I4=eye(m*n);%60x60
p=I4;%60x60
p11=2;%número de parámetros en el modelo AR(p)

wg=[...
0.0417771410819
0.0607327526904
0.0802072210065
0.1011474249135
0.1255943244188
0.1490939678064
0.1792778110118
0.2073378929935
0.2346812457027
0.2638262992382
0.2874380562301
0.3042015927428
0.3210900904786
0.3433870599539
0.3646789301295
0.3897292797403
0.4121809992766
0.4343919274996
0.4576039371666
0.4792028123558
0.4891948360894
0.4925829268489
0.4944643990539
0.4972455151476
0.4962180798019
0.4946155870404
0.4900210091871
0.4836995362948
0.4739458421301
0.4631267505220
0.4622823611477
0.4609818763533
0.4623828295527
0.4626169143686
0.4506274047320
0.4341976559052
0.4130215906737
0.3863259999982
0.3566454099371
0.3220103466064
0.2992731555646
0.2976832738058
0.2861549137108
0.2709928438105
0.2549741045690
0.2468318039499
0.2392210016892
```

```
0.2263833619683
0.2100240212499
0.1920641981328
0.1569055412596
0.1564799429007
0.1558071335628
0.1498794246465
0.1407164861421
0.1296650398429
0.1171006768309
0.1033659355783
0.0881248420945
0.1090406667123];%60x1

c0=[0 0 0 0 0 0 0 0 0 1];%1x10

%estas restricciones se sacan de valores observados censales
yt=[...
0.43367481329542
0.67055545003731
0.69525931840888
0.58342892673092
0.28718364734908
0.15330723923449];

c=kron(I3,c0);%6x60

%pia debe de ser de 60x60
pia=eye(m*n,m*n);
for j=1:m*n
        for i=1:(m*n)
            if i-j==1
                pia(i,j)=-1.88081476495;%se cambian los signos de los parámetros
            end
            if i-j==2
                pia(i,j)=0.883191142012;%se cambian los signos de los parámetros
            end
        end
end

r=[...
0
0
0.00287736397245
0.00393112859502
0.00619306132314
0.00220681776651
0.00978346981019
0.0018280107086
0.0030536498718
0.00555333775897
-0.00150194535829
-0.00340709690635
0.00280598846618
0.00814422899897
0.00241540137393
0.00711217311936
0.00125361637646
0.00336126587763
0.00462779361697
0.00218567336273
```

```
-0.00794514493964
-0.0042742647055
5.97032263272e-05
0.00229445036414
-0.00230204960823
0.000484130297357
-0.0020038773057
-0.0010991077449
-0.00302117281884
-0.00107844129212
0.00981150028529
0.000543829620547
0.00364499660706
9.55713097851e-05
-0.0110968990487
-0.00476985949061
-0.00563363925236
-0.00701158240622
-0.0051852246009
-0.00757390512763
0.00861740809628
0.0192025898257
-0.0094167829052
-0.00430031240938
-0.00198375223301
0.00661122259867
0.000166970958745
-0.0055473672795
-0.00448327880398
-0.0030123020389
-0.0188402832292
0.0309990826857
7.49307096932e-05
-0.00496323317052
-0.00357146847032
-0.00262442474741
-0.00249569050771
-0.00235973177302
-0.00286505523338
0.0345860412306
];%60X1

rt=r';%1x60
sum=zeros(k1,k1);
suma=zeros(k1,k1);
for i=1:m*n
                    sum=r(i,1)*rt(1,i);
                    suma=suma+sum;
end

sigma=[(m*n-p11)^(-1)]*suma;%

a=inv(pia)*(inv(pia))'*c'*pinv([c*inv(pia)*(inv(pia))'*c']);% 60x6
%   60x60*60x60     *60x6*    6x60*  60x60*60x60     *60x6

COVzgz=(I21-a*c)*inv(pia)*kron(p,sigma)*(inv(pia))';%60x60
%      60x60-60x6*6x60 *60x60  *60x60          *60x60

zg=wg+a*(yt-c*wg);
%60x1=60x1+60x1
```

```
var1=diag(COVzgz);
for i=1:size(var1)
                    if var1(i)<0.00000000000000000001 %20 digitos
                        var1(i)=0;
                    else
                        var1(i)=var1(i);
                    end
end
se1=var1.^(1/2);

for i=1941:2000
  ano(i)=i;
end

anos=(ano(:,1941:2000))';
anosa=[1950 1960 1970 1980 1990 2000]';

figure
plot(anos,wg,'b');
grid on;
hold on;
plot(anos,zg,'b--');
hold on;
plot(anosa,yt,'b*');
hold on;
legend('wt','zt','censal data',-1);%pone las leyendas fuera de la gráfica
xlabel('year')
ylabel('population')
AXIS([1940 2000 0 0.8])

at=pia*(zg-wg);%para obtener residuales que permitan ver si se tiene ruido blanco
%60x1=60x60*(60*1-60*1)

%no se tiene ruido blanco por lo que se pasa a la segunda etapa.

%%%%%%%para segunda etapa

e=[...
0
0
0
0
0.000130434422916
6.90082340486e-05
-4.26305925262e-06
-9.04086575297e-05
-0.000190591392871
-0.000306119250397
-0.000438458306974
0.00114490745587
6.09600435916e-07
1.46852513722e-05
3.31223875972e-05
5.6205590765e-05
8.42674653569e-05
0.000117691944191
0.000156918094898
0.000202444464538
0.000254834004591
-0.000291572998734
0.000116777652824
```

```
0.000121336437489
0.000125660135077
0.00012975461416
0.000133622560163
0.000137263477003
0.00014067365726
0.000143846120345
0.000146770518274
-0.000182162821948
6.31644859574e-06
-3.1269328669e-05
-7.47138836234e-05
-0.000124581386063
-0.00018150221002
-0.000246179159297
-0.000319394399887
-0.000402017166791
-0.000495012320071
0.000708855642695
-0.000142449054701
-0.000116945605759
-8.50073834644e-05
-4.61335403272e-05
2.49319083378e-07
5.47928424664e-05
0.000118233398564
0.000191399373157
0.000275219363722
-0.000673232841658
2.10159700125e-05
1.86286251801e-05
1.43936967276e-05
8.21905245327e-06
-7.66967414041e-09
-1.04201885148e-05
-2.31749155055e-05
-3.84525117917e-05];%60x1

et=e';
sum1=zeros(k1,k1);
suma1=zeros(k1,k1);
for i=1:m*n
                sum1=e(i,1)*et(1,i);
                suma1=suma1+sum1;
end

p12=4;%ORDEN DEL MODELO AUTOREGRESIVO

sigma1=[(m*n-p12)^(-1)]*suma1;% LOS et`s son los residuales del modelo de la segunda etapa

lambda=eye(m*n,m*n);% los valores primeros corresponden a los elementos autoregresivos
for j=1:m*n
                for i=1:(m*n)
                    if i-j==1
                            lambda(i,j)=-3.57159572908;%se cambian los signos de los parámetros
                    end
                    if i-j==2
                            lambda(i,j)=4.81948419167;%se cambian los signos de los parámetros
                    end
                    if i-j==3
                            lambda(i,j)=-2.91327830487;%se cambian los signos de los parámetros
```

```
                    end
                    if i-j==4
                              lambda(i,j)=0.665598886328;%se cambian los signos de los parámetros
                    end
               end
end

a2=inv(lambda)*kron(eye(m*n),sigma1)*(inv(lambda))'*c'*pinv([c*inv(lambda)*kron(eye(m*n),sigma1)*(inv(lambda))'*c']);
% 60x6

COVzgz2=(I21-a2*c)*inv(lambda)*kron(eye(m*n),sigma1)*(inv(lambda))';%60x60
%     60x60-60x6*6x60 *60x60  *60x60         *60x60

zg2=wg+a2*(yt-c*wg);%
%60x1=60x1+60x1

%%%para intervalos de confianza de las estimaciones intercensales con un nivel de confianza del 95%
var=diag(COVzgz2);
for i=1:size(var)
               if var(i)<0.00000000000000000001 %20 digitos
                    var(i)=0;
               else
                    var(i)=var(i);
               end
end
se=var.^(1/2);

for i=1:size(zg2)
               burS(i)=zg2(i)+1.96*se(i);% 95% nivel de confianza
               burI(i)=zg2(i)-1.96*se(i);% 95% nivel de confianza
end

for i=1:size(zg2)
               if i==10 | i==20 | i==30 | i==40 | i==50 | i==60
                    ls(i)=zg2(i);
                    li(i)=zg2(i);
               else
                    ls(i)=burS(i);
                    li(i)=burI(i);
               end
end


figure
anos=(ano(:,1941:2000))';
anosa=[1950 1960 1970 1980 1990 2000]';
plot(anos,wg,'r');
grid on;
hold on;
plot(anos,zg,'b--');
hold on;
plot(anos,zg2,'b-.');
hold on;
plot(anos,ls,'b:');
hold on;
plot(anos,li,'b:');
hold on;
plot(anosa,yt,'bo');
hold on;
legend('wccp','z','zccp','upper bound','lower bound','censal data',-1);%pone las leyendas fuera de la gráfica
xlabel('year')
```

```
ylabel('population')
AXIS([1940 2000 0 0.8])

figure
plot(anos,wg,'b');
grid on;
hold on;
plot(anos,zg2,'b-.');
hold on;
plot(anos,ls,'b:');
hold on;
plot(anos,li,'b:');
hold on;
plot(anosa,yt,'bo');
hold on;
legend('wccp','zccp','upper bound','lower bound','censal data',-1);%pone las leyendas fuera de la gráfica
xlabel('year')
ylabel('population')
AXIS([1940 2000 0 0.8])

figure
plot(anos,wg,'b');
grid on;
hold on;
plot(anos,zg2,'b-.');
hold on;
plot(anos,ls,'b:');
hold on;
plot(anos,li,'b:');
hold on;
plot(anosa,yt,'bo');
hold on;
%legend('zccp2','upper bound','lower bound','censal data',-1);%pone las leyendas fuera de la gráfica
xlabel('year')
ylabel('cumulative population growth rate')
AXIS([1940 2000 0 0.8])

%TABLA RESUMEN DE DATOS.
lista=[wg zg se1 zg2 se];%4x60
lista2=[anos wg zg zg2 se];%4x60

[anos li' zg2 ls']
```

2.5.6 Matlab routine: frp disagregation

```
%PRIMERA ETAPA
%ESTA DESAGREGACIÓN UNIVARIADA SIRVE PARA GENERAR LA SERIE UNIVARIADA DE FRP RESTRINGIDA
%TEMPORALMENTE

clear all
format long g

n=6;
m=10;
k1=1; % número de series
I21=eye(k1*m*n);%60x60
I3=eye(n);%6x6
I4=eye(m*n);%60x60
p=I4;%60x60
p11=11;%orden del modelo AR(p)
```

```
wgfrp=[...
0.1560121402952
0.2566270258136
0.3515422708668
0.4425733124610
0.5324565311196
0.6173537883797
0.7053291704342
0.7880166431590
0.8671394625458
0.9454927756789
1.0833359994375
1.1795140963906
1.2725662822742
1.3681035948952
1.4601659497644
1.5537562705952
1.6428257161956
1.7299223189747
1.8164759573862
1.9000477432629
2.0341091307568
2.1213455589324
2.2064862805707
2.2921629216547
2.3740341047540
2.4555380386137
2.5345631538168
2.6126419005136
2.6883789333040
2.7644589124555
2.8331157362229
2.8798962808000
2.9313768794510
2.9839618835504
3.0269440678675
3.0682230307532
3.1077803319310
3.1451858817432
3.1833716818960
3.2208789577556
3.2219441520634
3.2420994584924
3.2533880282079
3.2621785001161
3.2713341219886
3.2896804222757
3.3100064489639
3.3266932900999
3.3415933010925
3.3565709827337
3.3647052478974
3.3797039714552
3.3951391807750
3.4060539093785
3.4145194160252
3.4219404355350
3.4287575470243
3.4353857000510
3.4415697958129
```

```
3.4472745028195
];%60x1

c0=[0 0 0 0 0 0 0 0 0 1];%1x10

%estas restricciones se sacan de valores observados censales
ytfrp=[...
1.11534128973563
2.09140038094360
2.99659148034345
3.48229753787970
3.45191841379543
3.49154107534030];

c=kron(I3,c0);%6x60

%pia debe de ser de 60x60
pia=eye(m*n,m*n);
for j=1:m*n
	for i=1:(m*n)
		if i-j==1
			pia(i,j)=-0.895030020019;%se cambian los signos de los parámetros
		end
		if i-j==10
			pia(i,j)=-0.372520136019;%se cambian los signos de los parámetros
		end
		if i-j==11
			pia(i,j)=0.498895757578;%se cambian los signos de los parámetros
		end
	end
end

r=[...
0
0
0
0
0
0
0
0
0
0
0
0.0045769562924
0.00229753492991
0.00440138061333
-0.000207281451396
0.00160387948531
-0.0056106101615
-0.00674916896931
-0.00747821047126
-0.0116979304441
0.0151926442391
-0.00423414829926
-0.00297205477594
-0.00128664655592
-0.000855110883777
0.000809273425349
0.00326688947221
0.0059731708361
```

```
0.00689972727122
0.0116019903127
-0.0112249376514
-0.00615497112248
0.00111183392203
0.00452811354602
-0.000464986470594
0.000168723464205
0.0018423352868
0.00251794237813
0.00679681673627
0.00889973227313
-0.0213988262837
0.00495239654037
-0.00681724993721
-0.0107186364056
-0.0073938349465
0.00113544148478
0.00370707344736
0.00130718479501
-0.000489573765428
0.00052957099686
0.00836961651301
0.00440387107096
0.00805466748087
0.00379826822756
0.000253235359345
-0.00488875996544
-0.00535517493725
-0.00262974705606
-0.000832258459736
0.000459848613598
];%60X1

rt=r';%1x60
sum=zeros(k1,k1);
suma=zeros(k1,k1);
for i=1:m*n
        sum=r(i,1)*rt(1,i);
        suma=suma+sum;
end

sigma=[(m*n-p11)^(-1)]*suma;%


a=inv(pia)*(inv(pia))'*c'*pinv([c*inv(pia)*(inv(pia))'*c']);% 60x6
%   60x60*60x60     *60x6*    6x60*  60x60*60x60     *60x6

COVzgz=(I21-a*c)*inv(pia)*kron(p,sigma)*(inv(pia))';%60x60
%     60x60-60x6*6x60 *60x60  *60x60          *60x60

zg=wgfrp+a*(ytfrp-c*wgfrp);
%60x1=60x1+60x1

var1=diag(COVzgz);
for i=1:size(var1)
        if var1(i)<0.00000000000000000001 %20 digitos
                var1(i)=0;
        else
                var1(i)=var1(i);
        end
```

```
end
se1=var1.^(1/2);

for i=1941:2000
    ano(i)=i;
end

anos=(ano(:,1941:2000))';
anosa=[1950 1960 1970 1980 1990 2000]';

figure
plot(anos,wgfrp,'r');
grid on;
hold on;
plot(anos,zg,'r--');
hold on;
plot(anosa,ytfrp,'ro');
hold on;
legend('wfrp','zfrp','censal data',-1);%pone las leyendas fuera de la gráfica
xlabel('year')
ylabel('population')
AXIS([1940 2000 0 4])

at=pia*(zg-wgfrp);%para obtener residuales que permitan ver si se tiene ruido blanco
%60x1=60x60*(60*1-60*1)

%no se tiene ruido blanco por lo que se pasa a la segunda etapa.
%%%%%%%para segunda etapa

e=[...
0
0
0
0
0
0
0
0
0
0
0
0
-0.00110970311522
-0.000566460565907
0.000369174610073
0.00174723079247
0.00361016173393
0.00598500765307
0.00887232672865
0.0122309014972
-0.0259572579319
0.000362173158
0.0015941063406
0.00295176073999
0.00439515927652
0.00586926273504
0.00729845403043
0.00857968227865
0.00957398578042
0.0100960634383
-0.0113216326761
```

```
0.00274019459186
0.00322732532007
0.0035227179913
0.00358555611957
0.00336581454562
0.00280122303
0.0018138182427
0.00030599680248
-0.00184402920633
-0.0031021510477
-0.00427387047055
-0.00426592162974
-0.00445496763823
-0.00484273378882
-0.00543793307007
-0.00625743521261
-0.00732744708943
-0.0086847465328
-0.0103780105125
0.0336585960341
-0.00367281338124
-0.00262684249143
-0.00185007801583
-0.0013115658831
-0.000989873322033
-0.0008722570638
-0.000953866193905
-0.00123700309702
-0.00173045950912
];%60x1

et=e';
sum1=zeros(k1,k1);
suma1=zeros(k1,k1);
for i=1:m*n
        sum1=e(i,1)*et(1,i);
        suma1=suma1+sum1;
end

p12=12;%ORDEN DEL MODELO AUTOREGRESIVO

sigma1=[(m*n-p12)^(-1)]*suma1;% LOS et`s son los residuales del modelo de la segunda etapa
lambda=eye(m*n,m*n);% los valores primeros corresponden a los elementos autoregresivos
for j=1:m*n
        for i=1:(m*n)
                if i-j==1
                        lambda(i,j)=-1.66270893104;%se cambian los signos de los parámetros
                end
                if i-j==2
                        lambda(i,j)=0.697352293649;%se cambian los signos de los parámetros
                end
                if i-j==10
                        lambda(i,j)=-0.921748876469;%se cambian los signos de los parámetros
                end
                if i-j==11
                        lambda(i,j)=1.64630576945;%se cambian los signos de los parámetros
                end
                if i-j==12
                        lambda(i,j)=-0.745933506302;%se cambian los signos de los parámetros
                end
        end
```

```
end

a2=inv(lambda)*kron(eye(m*n),sigma1)*(inv(lambda))'*c'*pinv([c*inv(lambda)*kron(eye(m*n),sigma1)*(inv(lambda))'*c']);
% 60x6

COVzgz2=(I21-a2*c)*inv(lambda)*kron(eye(m*n),sigma1)*(inv(lambda))';%60x60
%      60x60-60x6*6x60 *60x60  *60x60          *60x60

zg2frp=wgfrp+a2*(ytfrp-c*wgfrp);%
%60x1=60x1+60x1

%%%para intervalos de confianza de las estimaciones intercensales con un nivel de confianza del 95%
var=diag(COVzgz2);
for i=1:size(var)
        if var(i)<0.00000000000000000001 %20 digitos
                var(i)=0;
        else
                var(i)=var(i);
        end
end
se=var.^(1/2);

for i=1:size(zg2frp)
        burS(i)=zg2frp(i)+1.96*se(i);% 95% nivel de confianza
        burI(i)=zg2frp(i)-1.96*se(i);% 95% nivel de confianza
end

for i=1:size(zg2frp)
        if i==10 | i==20 | i==30 | i==40 | i==50 | i==60
                lsfrp(i)=zg2frp(i);
                lifrp(i)=zg2frp(i);
        else
                lsfrp(i)=burS(i);
                lifrp(i)=burI(i);
        end
end

figure
anos=(ano(:,1941:2000))';
anosa=[1950 1960 1970 1980 1990 2000]';
plot(anos,wgfrp,'r');
grid on;
hold on;
plot(anos,zg,'r--');
hold on;
plot(anos,zg2frp,'r-.');
hold on;
plot(anos,lsfrp,'r:');
hold on;
plot(anos,lifrp,'r:');
hold on;
plot(anosa,ytfrp,'ro');
hold on;
legend('wfrp','zfrp','zfrp2','upper bound','lower bound','censal data',-1);%pone las leyendas fuera de la gráfica
AXIS([1940 2000 0 4])

figure
plot(anos,wgfrp,'r');
grid on;
hold on;
plot(anos,zg2frp,'r-.');
```

```
hold on;
plot(anos,lsfrp,'r:');
hold on;
plot(anos,lifrp,'r:');
hold on;
plot(anosa,ytfrp,'ro');
hold on;
%legend('wfrp','zfrp','Upper bound','Lower bound','Census data',-1);%pone las leyendas fuera de la gráfica
xlabel('year')
ylabel('cumulative population growth rate')
AXIS([1940 2000 0 4])

lista2=[anos wgfrp zg zg2frp se];%4x60
[anos lifrp' zg2frp lsfrp']
```

2.5.7 Matlab routine: srp disagregation

```
%PRIMERA ETAPA
%ESTA DESAGREGACIÓN UNIVARIADA SIRVE PARA GENERAR LA SERIE UNIVARIADA DE SRP RESTRINGIDA
%TEMPORALMENTE

clear all
format long g

n=6;
m=10;
k1=1; % número de series
I21=eye(k1*m*n);%60x60
I3=eye(n);%6x6
I4=eye(m*n);%60x60
p=I4;%60x60
p11=2;%orden del modelo AR(p)

wgsrp=[...
0.0396728506784
0.0569667636788
0.0747779301325
0.0940534318927
0.1168344836528
0.1386660186500
0.1671805573719
0.1935687315758
0.2192379646691
0.2467068448563
0.3512276044285
0.4257974559240
0.4992784025996
0.5770212386832
0.6528090824648
0.7314673208795
0.8067759681243
0.8811553251796
0.9559218588620
1.0285378496735
1.1621701849569
1.2491452827501
1.3340472519583
1.4195050843306
1.5011748160061
1.5824928432432
```

```
1.6613458334293
1.7392648200040
1.8148531829814
1.8907945019615
2.0651614396283
2.1753190850527
2.2859174618328
2.3939488444526
2.4892188148882
2.5798131675587
2.6659963280299
2.7475844110163
2.8277164543504
2.9051322253030
2.9492903630511
3.0064182264733
3.0542771152070
3.0992314453234
3.1441632374992
3.1979110341448
3.2533137879995
3.3047845579183
3.3541936000649
3.4034259827249
3.4401791642344
3.4820738390580
3.5244239223502
3.5622867543545
3.5977391868414
3.6321936295284
3.6660998518570
3.6998824310170
3.7332963991612
3.7663165875700];%60x1

c0=[0 0 0 0 0 0 0 0 0 1];%1x10

%estas restricciones se sacan de valores observados censales
ytsrp=[...
0.41655535891120
1.21989048735370
2.12292706984905
3.16655080542718
3.49877341378654
3.81058316009082];


c=kron(I3,c0);%6x60

%pia debe de ser de 60x60
pia=eye(m*n,m*n);
for j=1:m*n
        for i=1:(m*n)
                if i-j==1
                        pia(i,j)=-1.23649068546;%se cambian los signos de los parámetros
                end
                if i-j==2
                        pia(i,j)=0.410687416085;%se cambian los signos de los parámetros
                end
        end
end
```

```
r=[...
0
0
0.0169169097277
0.0168158535801
0.0170531337135
0.0111566228844
0.0132202793214
0.00327860255424
-0.00312609387396
-0.00872921601002
0.0592859971497
0.0019997149449
0.0115808842104
0.0138384115382
0.00787943850944
0.00849750133707
0.0010329623507
-0.00234454792896
-0.00579250273993
-0.0124126637977
0.0447067088225
-0.0211925543064
-0.00638766741908
-0.00754164923288
-0.0138864847408
-0.015485682689
-0.0204399685119
-0.0231516574201
-0.0277395864157
-0.029129121325
0.0668925360287
-0.0224448557719
0.0089075389605
0.0112323426723
0.00462328447873
0.00855095824338
0.00910635250833
0.00910720803466
0.0120329432507
0.0126673870358
-0.0166858599206
0.00744119998227
-0.00690213245646
-0.00688128822966
-0.0066331696075
0.00171129244406
0.00122277780216
-0.00122775529926
0.000167211113977
0.0026522778214
-0.00767474218181
0.00275269105355
0.00236442266632
-0.000798495627647
-0.000522801532011
-4.8047388284e-05
0.000124535768764
0.000382502519475
7.62282661679e-05
```

```
-0.000410012278275];%60X1

rt=r';%1x60
sum=zeros(k1,k1);
suma=zeros(k1,k1);
for i=1:m*n
        sum=r(i,1)*rt(1,i);
        suma=suma+sum;
end

sigma=[(m*n-p11)^(-1)]*suma;%

a=inv(pia)*(inv(pia))'*c'*pinv([c*inv(pia)*(inv(pia))'*c']);% 60x6
%   60x60*60x60     *60x6*    6x60*  60x60*60x60     *60x6

COVzgz=(I21-a*c)*inv(pia)*kron(p,sigma)*(inv(pia))';%60x60
%      60x60-60x6*6x60 *60x60  *60x60          *60x60

zg=wgsrp+a*(ytsrp-c*wgsrp);
%60x1=60x1+60x1

var1=diag(COVzgz);
for i=1:size(var1)
        if var1(i)<0.00000000000000000001 %20 digitos
                var1(i)=0;
        else
                var1(i)=var1(i);
        end
end
se1=var1.^(1/2);

for i=1941:2000
   ano(i)=i;
end

anos=(ano(:,1941:2000))';
anosa=[1950 1960 1970 1980 1990 2000]';

figure
plot(anos,wgsrp,'g');
grid on;
hold on;
plot(anos,zg,'g--');
hold on;
plot(anosa,ytsrp,'go');
hold on;
legend('wsrp','zsrp','censal data',-1);%pone las leyendas fuera de la gráfica
xlabel('year')
ylabel('population')
AXIS([1940 2000 0 4])

at=pia*(zg-wgsrp);%para obtener residuales que permitan ver si se tiene ruido blanco
%60x1=60x60*(60*1-60*1)

%no se tiene ruido blanco por lo que se pasa a la segunda etapa.
%%%%%%%para segunda etapa

e=[...
0
0
0
```

```
0
0
0
0
0
0
0
0
0
0
9.9609730555e-05
-0.000875030600187
-0.000417075300701
0.000601639917723
0.00147875699172
0.0014324509377
-0.00065670251763
-0.00651910966889
0.00339858922577
0.00197834767947
0.00163735435792
0.00202432721027
0.00284701169487
0.00377132162464
0.00427794268025
0.00346562507354
-0.000209104556746
-0.00925358368203
0.00509330868901
0.00248234483635
0.00153812120786
0.00165619511084
0.0023811547539
0.00328528376271
0.00380923301771
0.00305690487514
-0.000464841277893
-0.00915981604439
0.00318365996986
-7.907688779e-05
-0.00238030987698
-0.00423695113522
-0.00588010278295
-0.00717502056089
-0.00746638741695
-0.00532059313636
0.00185028797258
0.0182698764591
-0.00933411955046
-0.00236972992038
0.000622705049089
0.00136182906143
0.000963091475419
0.000159956837297
-0.00048526708434
-0.000383044104411
0.00127684087888];%60X1

et=e';
sum1=zeros(k1,k1);
suma1=zeros(k1,k1);
```

```
for i=1:m*n
	sum1=e(i,1)*et(1,i);
	suma1=suma1+sum1;
end

p12=13;%ORDEN DEL MODELO AUTOREGRESIVO

sigma1=[(m*n-p12)^(-1)]*suma1;% LOS et`s son los residuales del modelo de la segunda etapa

lambda=eye(m*n,m*n);% los valores primeros corresponden a los elementos autoregresivos
for j=1:m*n
	for i=1:(m*n)
		if i-j==1
			lambda(i,j)=-2.37484776414;%se cambian los signos de los parámetros
		end
		if i-j==2
			lambda(i,j)=2.04680194407;%se cambian los signos de los parámetros
		end
		if i-j==3
			lambda(i,j)=-0.649528723372;%se cambian los signos de los parámetros
		end
		if i-j==10
			lambda(i,j)=-0.851387780705;%se cambian los signos de los parámetros
		end
		if i-j==11
			lambda(i,j)=2.04183695069;%se cambian los signos de los parámetros
		end
		if i-j==12
			lambda(i,j)=-1.77747787111;%se cambian los signos de los parámetros
		end
		if i-j==13
			lambda(i,j)=0.571902121602;%se cambian los signos de los parámetros
		end
	end
end

a2=inv(lambda)*kron(eye(m*n),sigma1)*(inv(lambda))'*c'*pinv([c*inv(lambda)*kron(eye(m*n),sigma1)*(inv(lambda))'*c']);%
60x6
%   60x60*60x60     *60x6*    6x60*  60x60*60x60     *60x6

COVzgz2=(I21-a2*c)*inv(lambda)*kron(eye(m*n),sigma1)*(inv(lambda))';%60x60
%      60x60-60x6*6x60 *60x60  *60x60          *60x60

zg2srp=wgsrp+a2*(ytsrp-c*wgsrp);%
%60x1=60x1+60x1

%%%para intervalos de confianza de las estimaciones intercensales con un nivel de confianza del 95%
var=diag(COVzgz2);
for i=1:size(var)
	if var(i)<0.00000000000000000001 %20 digitos
		var(i)=0;
	else
		var(i)=var(i);
	end
end
se=var.^(1/2);

for i=1:size(zg2srp)
	burS(i)=zg2srp(i)+1.96*se(i);% 95% nivel de confianza
	burI(i)=zg2srp(i)-1.96*se(i);% 95% nivel de confianza
end
```

```
for i=1:size(zg2srp)
        if i==10 | i==20 | i==30 | i==40 | i==50 | i==60
                lssrp(i)=zg2srp(i);
                lisrpsrp(i)=zg2srp(i);
        else
                lssrp(i)=burS(i);
                lisrpsrp(i)=burI(i);
        end
end

figure
anos=(ano(:,1941:2000))';
anosa=[1950 1960 1970 1980 1990 2000]';
plot(anos,wgsrp,'r');
grid on;
hold on;
plot(anos,zg,'g--');
hold on;
plot(anos,zg2srp,'g-.');
hold on;
plot(anos,lssrp,'g:');
hold on;
plot(anos,lisrpsrp,'g:');
hold on;
plot(anosa,ytsrp,'go');
hold on;
legend('wsrp','zsrp','zsrp2','upper bound','lower bound','censal data',-1);%pone las leyendas fuera de la gráfica
xlabel('year')
ylabel('population')
AXIS([1940 2000 0 4])

figure
plot(anos,wgsrp,'g');
grid on;
hold on;
plot(anos,zg2srp,'g-.');
hold on;
plot(anos,lssrp,'g:');
hold on;
plot(anos,lisrpsrp,'g:');
hold on;
plot(anosa,ytsrp,'go');
hold on;
%legend('wsrp','zsrp','Upper bound','Lower bound','Census data',-1);%pone las leyendas fuera de la gráfica
xlabel('year')
ylabel('cumulative population growth rate')
AXIS([1940 2000 0 4])

lista2=[anos wgsrp zg zg2srp se];%4x60
[anos lisrpsrp' zg2srp lssrp']
```

2.5.8 Matlab routine: trp disagregation

```
%PRIMERA ETAPA
%ESTA DESAGREGACIÓN UNIVARIADA SIRVE PARA GENERAR LA SERIE UNIVARIADA DE TRP RESTRINGIDA
TEMPORALMENTE

clear all
format long g
```

```
n=6;
m=10;
k1=1; % número de series
I21=eye(k1*m*n);%60x60
I3=eye(n);%6x6
I4=eye(m*n);%60x60
p=I4;%60x60
p11=2;%orden del modelo AR(p)


wgtrp=[...
0.0193533150192
0.0202734623072
0.0213941410617
0.0236529016411
0.0290814815098
0.0332001692291
0.0436389908778
0.0515586041364
0.0583547841616
0.0665311257464
0.0966181451491
0.1183114127470
0.1402846726937
0.1678232611403
0.1945291343320
0.2251701792640
0.2534026836578
0.2815914467195
0.3109876622509
0.3389878972398
0.3991441292693
0.4385509985444
0.4779116749057
0.5197186885091
0.5594336811702
0.6003997408680
0.6403752454494
0.6807985940148
0.7201786297201
0.7611202768893
0.8470221198703
0.9048922977532
0.9672502418236
1.0305224737756
1.0840284994843
1.1356418479143
1.1853455140150
1.2327116048148
1.2806729046363
1.3277758751606
1.4010435303494
1.4815448203444
1.5511906535025
1.6164873100886
1.6804538852164
1.7520464815427
1.8242251969995
1.8915012478069
1.9558258232738
```

```
2.0191573904059
2.0592128926176
2.1041165930374
2.1493892028355
2.1900922209564
2.2283055107612
2.2654445486410
2.3019621505976
2.3382858094458
2.3741733522494
2.4096022467964];%60x1

c0=[0 0 0 0 0 0 0 0 0 1];%1x10

%estas restricciones se sacan de valores observados censales
yttrp=[...
0.23637963980233
0.53034053492195
0.99325284477627
1.58919445528423
1.98293602726556
2.45386881931731];

c=kron(I3,c0);%6x60

%pia debe de ser de 60x60
pia=eye(m*n,m*n);
for j=1:m*n
        for i=1:(m*n)
                if i-j==1
                        pia(i,j)=-1.31776254393;%se cambian los signos de los parámetros
                end
                if i-j==2
                        pia(i,j)=0.402959105151;%se cambian los signos de los parámetros
                end
        end
end

r=[...
0
0
0.00183280421041
0.00219546392596
0.00401067949727
0.000509392079978
0.00605373647558
-4.51270533986e-05
-0.00165847497703
-0.00167370052417
0.0177132150988
0.00016254892436
0.00256525150855
0.00657548366889
0.00233575349158
0.0051850303828
-7.1285147506e-05
-0.000777418315493
-0.00134447008959
-0.00506152676667
0.0255684813508
-0.00761221623242
```

```
-0.000653467631001
0.000346904784188
-0.00408128358698
-0.00359987294736
-0.00667444865213
-0.00755535247884
-0.0105222861156
-0.0104155129173
0.0321402102032
-0.0119761229324
0.00343891966497
0.00255644190616
-0.00748084786918
-0.00615482553778
-0.00815153439294
-0.010693542931
-0.0102808521679
-0.0123939314507
0.0130976493519
0.0110774139291
-0.000697725746723
0.000497404770694
0.00182920965066
0.0109073116365
0.0101165695756
0.00686426772879
0.00750792211497
0.00923324582265
-0.0120271570289
0.0020141418043
0.000847798094834
-0.00321891659556
-0.00339313173966
-0.00297994527564
-0.00254349742231
-0.00167241456083
-0.000979531645083
1.42133301061e-05
];%60X1

rt=r';%1x60
sum=zeros(k1,k1);
suma=zeros(k1,k1);
for i=1:m*n
        sum=r(i,1)*rt(1,i);
        suma=suma+sum;
end

sigma=[(m*n-p11)^(-1)]*suma;%


a=inv(pia)*(inv(pia))'*c'*pinv([c*inv(pia)*(inv(pia))'*c']);% 60x6
%   60x60*60x60     *60x6*    6x60*  60x60*60x60     *60x6

COVzgz=(I21-a*c)*inv(pia)*kron(p,sigma)*(inv(pia))';%60x60
%      60x60-60x6*6x60 *60x60  *60x60          *60x60

zg=wgtrp+a*(yttrp-c*wgtrp);
%60x1=60x1+60x1

var1=diag(COVzgz);
```

```
for i=1:size(var1)
        if var1(i)<0.00000000000000000001 %20 digitos
                var1(i)=0;
        else
                var1(i)=var1(i);
        end
end
se1=var1.^(1/2);

for i=1941:2000
   ano(i)=i;
end

anos=(ano(:,1941:2000))';
anosa=[1950 1960 1970 1980 1990 2000]';

figure
plot(anos,wgtrp,'m');
grid on;
hold on;
plot(anos,zg,'m--');
hold on;
plot(anosa,yttrp,'mo');
hold on;
legend('wtrp','ztrp','censal data',-1);%pone las leyendas fuera de la gráfica
xlabel('year')
ylabel('population')
AXIS([1940 2000 0 2.5])

at=pia*(zg-wgtrp);%para obtener residuales que permitan ver si se tiene ruido blanco
%60x1=60x60*(60*1-60*1)

%no se tiene ruido blanco por lo que se pasa a la segunda etapa.
%%%%%%%para segunda etapa

e=[...
0
0
0
0.00241976916304
0.003026278603
0.0034511377886
0.0036479361844
0.0033740039079
0.00204418260269
-0.00160829969387
-0.0102527069553
0.00622277872682
0.00317816528399
0.00205333996982
0.00183184433436
0.0019913701061
0.00220064390617
0.00212177272311
0.00120780109293
-0.00161746523706
-0.00857577312258
0.00528593953021
0.00297271596765
0.00225235773578
0.00227694956003
```

```
0.0025960179435
0.00289122030632
0.00278341964624
0.00160444339482
-0.0019938446408
-0.0108109801027
0.00663431773546
0.00366557537611
0.00269244520892
0.00263190612722
0.0028890205675
0.00300821560584
0.00240078535133
-3.40140876506e-06
-0.00637417179058
-0.0212135501328
0.00598393752148
0.000323328949892
-0.00221310313627
-0.00328439603151
-0.00364825505708
-0.00358726951593
-0.00306851445125
-0.00172529006632
0.00134914491529
0.00811052059482
-0.00357167842635
-0.000885122627847
0.000409144551053
0.00104015857685
0.00133960768695
0.00142549144351
0.00126610777804
0.000659865902062
-0.00088037340762];%60x1

et=e';
sum1=zeros(k1,k1);
suma1=zeros(k1,k1);
for i=1:m*n
        sum1=e(i,1)*et(1,i);
        suma1=suma1+sum1;
end

p12=3;%ORDEN DEL MODELO AUTOREGRESIVO

sigma1=[(m*n-p12)^(-1)]*suma1;% LOS et`s son los residuales del modelo de la segunda etapa

lambda=eye(m*n,m*n);% los valores primeros corresponden a los elementos autoregresivos
for j=1:m*n
        for i=1:(m*n)
                if i-j==1
                        lambda(i,j)=-2.51368229239;%se cambian los signos de los parámetros
                end
                if i-j==2
                        lambda(i,j)=2.17638899568;%se cambian los signos de los parámetros
                end
                if i-j==3
                        lambda(i,j)=-0.659280004046;%se cambian los signos de los parámetros
                end
        end
```

```
end

a2=inv(lambda)*kron(eye(m*n),sigma1)*(inv(lambda))'*c'*pinv([c*inv(lambda)*kron(eye(m*n),sigma1)*(inv(lambda))'*c']);
% 60x6

COVzgz2=(I21-a2*c)*inv(lambda)*kron(eye(m*n),sigma1)*(inv(lambda))';%60x60
%     60x60-60x6*6x60 *60x60  *60x60         *60x60

zg2trp=wgtrp+a2*(yttrp-c*wgtrp);%
%60x1=60x1+60x1

%%%para intervalos de confianza de las estimaciones intercensales con un nivel de confianza del 95%
var=diag(COVzgz2);
for i=1:size(var)
        if var(i)<0.00000000000000000001 %20 digitos
                var(i)=0;
        else
                var(i)=var(i);
        end
end
se=var.^(1/2);


for i=1:size(zg2trp)
        burS(i)=zg2trp(i)+1.96*se(i);% 95% nivel de confianza
        burI(i)=zg2trp(i)-1.96*se(i);% 95% nivel de confianza
end

for i=1:size(zg2trp)
        if i==10 | i==20 | i==30 | i==40 | i==50 | i==60
                lstrp(i)=zg2trp(i);
                litrp(i)=zg2trp(i);
        else
                lstrp(i)=burS(i);
                litrp(i)=burI(i);
        end
end

figure
anos=(ano(:,1941:2000))';
anosa=[1950 1960 1970 1980 1990 2000]';
plot(anos,wgtrp,'m');
grid on;
hold on;
plot(anos,zg,'m--');
hold on;
plot(anos,zg2trp,'m-.');
hold on;
plot(anos,lstrp,'m:');
hold on;
plot(anos,litrp,'m:');
hold on;
plot(anosa,yttrp,'mo');
hold on;
legend('wtrp','ztrp','ztrp2','upper bound','lower bound','censal data',-1);%pone las leyendas fuera de la gráfica
xlabel('year')
ylabel('population')
AXIS([1940 2000 0 2.5])

figure
plot(anos,wgtrp,'m');
```

```
grid on;
hold on;
plot(anos,zg2trp,'m-.');
hold on;
plot(anos,lstrp,'m:');
hold on;
plot(anos,litrp,'m:');
hold on;
plot(anosa,yttrp,'mo');
hold on;
%legend('wtrp','ztrp','Upper bound','Lower bound','Census data',-1);%pone las leyendas fuera de la gráfica
xlabel('year')
ylabel('cumulative population growth rate')
AXIS([1940 2000 0 2.5])

lista2=[anos wgtrp zg zg2trp se];
[anos litrp' zg2trp lstrp']
```

2.5.9 Matlab routine: multiple unrestricted forecast

```
clear all
format long
% Observaciones de las poblaciones de los 4 anillos (hay 4 series).
% Vienen de las desagregaciones (o series) univariadas

dccp=[...
     0.0428571124912705
     0.0654452184242305
     0.0924562340796012
      0.125724105495432
      0.167527221242493
       0.21295372385507
      0.268551329344816
      0.323958773121036
      0.378771982149694
       0.43367481329542
      0.479697714779279
       0.51427105684611
      0.543598219094641
      0.572715868286858
      0.595467444473393
      0.617312381345399
      0.632929435839928
      0.645932113817305
      0.658908578042359
       0.67055545003731
      0.672042898208078
      0.669304511159965
      0.668083672375116
      0.671127765669458
      0.673770781614496
      0.679016661589468
      0.683990093582827
      0.689322725727203
      0.692559236068027
       0.69525931840888
      0.707654225957997
      0.718550132428869
      0.730423481592994
      0.738832698103948
```

```
        0.732272674228469
        0.718213002840251
        0.696172227236794
        0.665339052030272
        0.628342031736926
         0.58342892673092
        0.547779820634428
        0.531065444450309
        0.502692893559484
        0.469506327529084
        0.434842170281281
        0.407986422176029
        0.382115379065324
        0.351937586895641
         0.31955153770287
        0.287183647349079
        0.239443513236358
        0.228371189206936
        0.218999808631436
        0.206251009595173
        0.192004220239608
        0.177413017251766
        0.162624313020328
        0.147733299009708
        0.132153392886408
        0.153307239234489];%60x1

dfrp=[...
         0.16316460031604
        0.276424222131035
        0.388108147566723
        0.498864760830923
        0.610369581103869
        0.717724891854865
        0.827815095882657
        0.930828261370753
         1.02659462490624
         1.11534128973563
         1.26038787642781
         1.36130289665109
         1.45740099251199
         1.55502750698674
         1.64882956308723
         1.74420000542032
         1.83515676894838
         1.92385794303019
         2.01071247400367
          2.0914003809436
         2.22242901238638
         2.30748877649783
         2.39214115758094
         2.47961522397262
         2.56587391971553
         2.65429898746213
         2.74227325914061
         2.83025513753562
         2.91504675164338
         2.99659148034345
         3.06877450381635
         3.11867446614995
         3.17387625461022
```

```
        3.23128375983701
        3.28018961650168
         3.3279870362112
        3.37361488863536
        3.41500521156654
         3.4527653273371
         3.4822975378797
        3.46888237893943
        3.47128498939508
        3.46374435789168
        3.45400283932268
        3.44565028948863
        3.44772358181581
         3.4527967742773
        3.45465757424185
        3.45419599121261
        3.45191841379543
        3.44384761281425
        3.44556567532026
         3.4513859956498
         3.4563322822492
        3.46196237862256
        3.46885040846361
        3.47639206892501
        3.48378620400443
        3.48938527718803
         3.4915410753403];%60x1

dsrp=[...
      0.0417424110806452
      0.0649035494205284
      0.0937216358996681
       0.129934037814172
       0.175505929302813
        0.22466829914738
       0.282268196028657
        0.33538014496618
       0.380664959375905
         0.4165553589112
       0.518325848779415
       0.582646743012127
       0.643917180344942
       0.712962069933203
       0.787361184147107
       0.873063581892212
       0.962075509404751
        1.05270458748719
        1.14104410876664
         1.2198904873537
         1.3515670711352
        1.43098950914888
        1.50731443712541
        1.58796638948189
        1.67198890817306
        1.76366005597135
        1.85877132637253
        1.95429597923336
        2.04311545842176
        2.12292706984905
        2.29092092084004
        2.38782232751077
```

```
    2.48412164409739
    2.58297669976391
    2.67864238144689
    2.78044407834261
    2.88606082191049
     2.9893539847758
    3.08567637946021
    3.16655080542718
    3.19865859675592
    3.23096679207388
    3.24720043831566
    3.26042844947558
     3.2789314303863
    3.31439947243008
    3.35964988810498
    3.40669041041675
    3.45366999077481
    3.49877341378654
    3.52822305218727
    3.56011607000047
    3.59148411651178
    3.61949597687131
    3.64793899841854
    3.67894858566108
    3.71244482823601
    3.74718767960374
    3.78062809543648
    3.81058316009082];%60x1

dtrp=[...
  0.0223859787534498
  0.0310814930441202
  0.0452494700018725
  0.0654042677447738
  0.0924600897152863
   0.120377349589503
   0.155014569711499
   0.185759816807666
    0.21243634605981
    0.23637963980233
   0.277570399943542
   0.305989414481085
   0.331123849602794
   0.359321214238537
   0.385271512660418
   0.414701993629559
    0.44201169448415
   0.470059804638193
   0.500353146954112
    0.53034053492195
   0.593470295155128
     0.6366268794043
   0.680254052476483
   0.726583996526517
   0.770854643189619
   0.816249265846106
   0.860448523237257
   0.904905495946564
     0.9482388552107
    0.99325284477627
     1.0836188628547
```

```
          1.14666249847501
          1.21510996983552
          1.28533356991428
           1.3462143249873
          1.40473135258364
          1.45952278793639
          1.50847210859703
          1.55270540574428
          1.58919445528423
          1.64407956535622
          1.69876746457056
          1.73638508354356
          1.76523943458698
          1.79051024426197
          1.82350895356633
           1.8596098827564
          1.89564975093983
          1.93561653964985
          1.98293602726556
           2.0158561327011
          2.06166943800033
          2.11423664558733
          2.16659374679721
          2.21878148787695
          2.27039496539393
          2.32041214486033
          2.36817991553435
          2.41277271133225
          2.45386881931731];%60x1

Yt=[dccp dfrp dsrp dtrp];
%60x4

Ytps=Yt(1:2,:);% Presample (se toman las dos primeras observaciones)
Ytm1=(Ytps(1,:))';% en el tiempo menos 1
Yt0=(Ytps(2,:))';% en el tiempo 0

Yt=Yt(3:60,:);
T=max(size(Yt));% T=58, se han quitado dos observaciones

Yt1=(Yt(1,:))';%4x1
Yt2=(Yt(2,:))';
Yt3=(Yt(3,:))';
Yt4=(Yt(4,:))';
Yt5=(Yt(5,:))';
Yt6=(Yt(6,:))';
Yt7=(Yt(7,:))';
Yt8=(Yt(8,:))';
Yt9=(Yt(9,:))';
Yt10=(Yt(10,:))';
Yt11=(Yt(11,:))';
Yt12=(Yt(12,:))';
Yt13=(Yt(13,:))';
Yt14=(Yt(14,:))';
Yt15=(Yt(15,:))';
Yt16=(Yt(16,:))';
Yt17=(Yt(17,:))';
Yt18=(Yt(18,:))';
Yt19=(Yt(19,:))';
Yt20=(Yt(20,:))';
Yt21=(Yt(21,:))';
```

```
Yt22=(Yt(22,:))';
Yt23=(Yt(23,:))';
Yt24=(Yt(24,:))';
Yt25=(Yt(25,:))';
Yt26=(Yt(26,:))';
Yt27=(Yt(27,:))';
Yt28=(Yt(28,:))';
Yt29=(Yt(29,:))';
Yt30=(Yt(30,:))';
Yt31=(Yt(31,:))';
Yt32=(Yt(32,:))';
Yt33=(Yt(33,:))';
Yt34=(Yt(34,:))';
Yt35=(Yt(35,:))';
Yt36=(Yt(36,:))';
Yt37=(Yt(37,:))';
Yt38=(Yt(38,:))';
Yt39=(Yt(39,:))';
Yt40=(Yt(40,:))';
Yt41=(Yt(41,:))';
Yt42=(Yt(42,:))';
Yt43=(Yt(43,:))';
Yt44=(Yt(44,:))';
Yt45=(Yt(45,:))';
Yt46=(Yt(46,:))';
Yt47=(Yt(47,:))';
Yt48=(Yt(48,:))';
Yt49=(Yt(49,:))';
Yt50=(Yt(50,:))';
Yt51=(Yt(51,:))';
Yt52=(Yt(52,:))';
Yt53=(Yt(53,:))';
Yt54=(Yt(54,:))';
Yt55=(Yt(55,:))';
Yt56=(Yt(56,:))';
Yt57=(Yt(57,:))';
Yt58=(Yt(58,:))';% Yt

%Reacomodo de la matriz Z

Z=[...
1     1     1     1     1     1     1     1     1     1     1     1     1     1
      1     1     1     1     1     1     1     1     1     1     1     1     1
      1     1     1     1     1     1     1     1     1     1     1     1     1
      1     1     1     1     1     1     1     1     1     1     1     1     1
      1     1     1     1     1;
Yt0   Yt1   Yt2   Yt3   Yt4   Yt5   Yt6   Yt7   Yt8   Yt9   Yt10  Yt11  Yt12  Yt13
      Yt14  Yt15  Yt16  Yt17  Yt18  Yt19  Yt20  Yt21  Yt22  Yt23  Yt24  Yt25  Yt26
      Yt27  Yt28  Yt29  Yt30  Yt31  Yt32  Yt33  Yt34  Yt35  Yt36  Yt37  Yt38  Yt39
      Yt40  Yt41  Yt42  Yt43  Yt44  Yt45  Yt46  Yt47  Yt48  Yt49  Yt50  Yt51  Yt52
      Yt53  Yt54  Yt55  Yt56  Yt57;
Ytm1  Yt0   Yt1   Yt2   Yt3   Yt4   Yt5   Yt6   Yt7   Yt8   Yt9   Yt10  Yt11  Yt12
      Yt13  Yt14  Yt15  Yt16  Yt17  Yt18  Yt19  Yt20  Yt21  Yt22  Yt23  Yt24  Yt25
      Yt26  Yt27  Yt28  Yt29  Yt30  Yt31  Yt32  Yt33  Yt34  Yt35  Yt36  Yt37  Yt38
      Yt39  Yt40  Yt41  Yt42  Yt43  Yt44  Yt45  Yt46  Yt47  Yt48  Yt49  Yt50  Yt51
      Yt52  Yt53  Yt54  Yt55  Yt56
];%9x58 %

ZZ=(Z*Z');%9x9
ZZI=(Z*Z')\eye(size(Z*Z'));%9x9
```

```
%Estimaciones del modelo VAR (las proporciona E-Views)

v=[...
0.0206055860253  0.0390102424871  -0.0362316007769 -0.0143857448846]';%1x4

A1=[...
1.51593527068      0.162174782139    0.574054627682    0.209032660882
-0.0876411981519 1.50477453653     0.290751238695    0.154752328409
-0.0477359681843 -0.167786556076   1.38687980955     -0.057715681123
0.173299266006    -0.0658436412857 -0.705029940596   1.30708304588]';%4x4

A2=[...
-0.619068171897   -0.192136306221   -0.564436153675   -0.261308899307
0.155309197298    -0.451053986565   -0.240609885874   -0.115520936845
-0.0306769499392 0.0988871247145  -0.398732858902   0.0667258776148
-0.154490707764   0.0850405564493  0.688028093304    -0.356638309654]';%4x4

%Matriz BM definida en  3.5.10 en Lutkepohl
%Componentes de la matriz BM

Uk=ones(1,1);
ck=zeros(1,4);
ck1=zeros(4,1);
ck2=zeros(4,4);
Ik=eye(4,4);

BM=[...
Uk  ck ck;
 v  A1 A2;
ck1 Ik ck2];%9x9

%Resultados en los pronósticos

 YtF1=v+A1*Yt58+A2*Yt57;
 YtF2=v+A1*YtF1+A2*Yt58;
 YtF3=v+A1*YtF2+A2*YtF1;
 YtF4=v+A1*YtF3+A2*YtF2;
 YtF5=v+A1*YtF4+A2*YtF3;
 YtF6=v+A1*YtF5+A2*YtF4;
 YtF7=v+A1*YtF6+A2*YtF5;
 YtF8=v+A1*YtF7+A2*YtF6;
 YtF9=v+A1*YtF8+A2*YtF7;
YtF10=v+A1*YtF9+A2*YtF8;
YtF11=v+A1*YtF10+A2*YtF9;
YtF12=v+A1*YtF11+A2*YtF10;
YtF13=v+A1*YtF12+A2*YtF11;
YtF14=v+A1*YtF13+A2*YtF12;
YtF15=v+A1*YtF14+A2*YtF13;
YtF16=v+A1*YtF15+A2*YtF14;
YtF17=v+A1*YtF16+A2*YtF15;
YtF18=v+A1*YtF17+A2*YtF16;
YtF19=v+A1*YtF18+A2*YtF17;
YtF20=v+A1*YtF19+A2*YtF18;

%Matriz de reacomodo de los pronósticos

YtF=[...
YtF1 YtF2 YtF3 YtF4 YtF5 YtF6 YtF7 YtF8 YtF9 YtF10 YtF11 YtF12 YtF13 YtF14 YtF15 YtF16 YtF17 YtF18 YtF19 YtF20
];

%Matriz de Covarianza de los residuales (la proporciona E-Views)
```

```
Sigmau=[...
6.63860345977e-05    -1.28700575957e-06    1.27587663047e-05    2.63052014903e-05
-1.28700575957e-06   0.000162216228908     0.000168990625456    5.66617489479e-05
1.27587663047e-05    0.000168990625456     0.000341721470062    0.000129974549723
2.63052014903e-05    5.66617489479e-05     0.000129974549723    8.59137111394e-05];%4x4

K=4;%Número de series
p=2;%Orden del modelo VAR
Kp=K*p;

%Estimated forecast MSE matrix for h=1

Sigmay1cg=((T+Kp+1)/T)*Sigmau;%cg = Con gorro (estimado)%4x4

    %Matrices intermedias para calcular Sigmay2cg
    fi1=A1;%Primera matriz de coeficientes del modelo VAR estimado     %4x4

    Sigmay2sg=Sigmau+fi1*Sigmau*fi1';%sg = Sin gorro      %4x4

Omega2=trace(BM'*ZZI*BM*Z*Z')*Sigmau+trace(BM')*Sigmau*(fi1)'+trace(BM)*fi1*Sigmau+trace(eye(Kp+1))*fi1*Sigmau*(fi
1)';     %4x4

%Estimated forecast MSE matrix for h=2

Sigmay2cg=Sigmay2sg+(1/T)*Omega2;
%4x4

Sencillo=[Sigmay2sg Omega2 Sigmay2cg]
%*************
%Genérico

f0=     eye(K,K);
f1=     A1;
f2=  f1*A1+    A2;
f3=  f2*A1+ f1*A2;
f4=  f3*A1+ f2*A2;
f5=  f4*A1+ f3*A2;
f6=  f5*A1+ f4*A2;
f7=  f6*A1+ f5*A2;
f8=  f7*A1+ f6*A2;
f9=  f8*A1+ f7*A2;
f10= f9*A1+ f8*A2;
f11=f10*A1+ f9*A2;
f12=f11*A1+f10*A2;
f13=f12*A1+f11*A2;
f14=f13*A1+f12*A2;
f15=f14*A1+f13*A2;
f16=f15*A1+f14*A2;
f17=f16*A1+f15*A2;
f18=f17*A1+f16*A2;
f19=f18*A1+f17*A2;
f20=f19*A1+f18*A2;

f=[...
 f0;
 f1;
 f2;
 f3;
 f4;
 f5;
 f6;
```

```
 f7;
 f8;
 f9;
f10;
f11;
f12;
f13;
f14;
f15;
f16;
f17;
f18;
f19;
f20];

%%%%%%%%%%%%%%% Pronósticos sin ser corregidos
%Assuming that the data are generated by Gaussian process, we get the following aproximate 95%
%interval forecasts

YtF=[...
YtF1 YtF2 YtF3 YtF4 YtF5 YtF6 YtF7 YtF8 YtF9 YtF10 YtF11 YtF12 YtF13 YtF14 YtF15 YtF16 YtF17 YtF18 YtF19 YtF20
];%4x20, cada matriz es de 4x1

%Generación de matriz MSE para pronósticos sin corregir para h=1 hasta 20

 Sigma0sg =   Sigmau;
 Sigma1sg =   Sigmau+ f1*Sigmau*f1';
 Sigma2sg = Sigma1sg+ f2*Sigmau*f2';
 Sigma3sg = Sigma2sg+ f3*Sigmau*f3';
 Sigma4sg = Sigma3sg+ f4*Sigmau*f4';
 Sigma5sg = Sigma4sg+ f5*Sigmau*f5';
 Sigma6sg = Sigma5sg+ f6*Sigmau*f6';
 Sigma7sg = Sigma6sg+ f7*Sigmau*f7';
 Sigma8sg = Sigma7sg+ f8*Sigmau*f8';
 Sigma9sg = Sigma8sg+ f9*Sigmau*f9';
 Sigma10sg= Sigma9sg+f10*Sigmau*f10';
 Sigma11sg=Sigma10sg+f11*Sigmau*f11';
 Sigma12sg=Sigma11sg+f12*Sigmau*f12';
 Sigma13sg=Sigma12sg+f13*Sigmau*f13';
 Sigma14sg=Sigma13sg+f14*Sigmau*f14';
 Sigma15sg=Sigma14sg+f15*Sigmau*f15';
 Sigma16sg=Sigma15sg+f16*Sigmau*f16';
 Sigma17sg=Sigma16sg+f17*Sigmau*f17';
 Sigma18sg=Sigma17sg+f18*Sigmau*f18';
 Sigma19sg=Sigma18sg+f19*Sigmau*f19';
 Sigma20sg=Sigma19sg+f20*Sigmau*f20';

%Generación de límite inferior de pronósticos sin corregir

li1=YtF1-1.96*sqrt(diag(Sigma1sg));
li2=YtF2-1.96*sqrt(diag(Sigma2sg));
li3=YtF3-1.96*sqrt(diag(Sigma3sg));
li4=YtF4-1.96*sqrt(diag(Sigma4sg));
li5=YtF5-1.96*sqrt(diag(Sigma5sg));
li6=YtF6-1.96*sqrt(diag(Sigma6sg));
li7=YtF7-1.96*sqrt(diag(Sigma7sg));
li8=YtF8-1.96*sqrt(diag(Sigma8sg));
li9=YtF9-1.96*sqrt(diag(Sigma9sg));
li10=YtF10-1.96*sqrt(diag(Sigma10sg));
li11=YtF11-1.96*sqrt(diag(Sigma11sg));
li12=YtF12-1.96*sqrt(diag(Sigma12sg));
```

```
li13=YtF13-1.96*sqrt(diag(Sigma13sg));
li14=YtF14-1.96*sqrt(diag(Sigma14sg));
li15=YtF15-1.96*sqrt(diag(Sigma15sg));
li16=YtF16-1.96*sqrt(diag(Sigma16sg));
li17=YtF17-1.96*sqrt(diag(Sigma17sg));
li18=YtF18-1.96*sqrt(diag(Sigma18sg));
li19=YtF19-1.96*sqrt(diag(Sigma19sg));
li20=YtF20-1.96*sqrt(diag(Sigma20sg));

%Generación de límite superior de pronósticos

ls1=YtF1+1.96*sqrt(diag(Sigma1sg));
ls2=YtF2+1.96*sqrt(diag(Sigma2sg));
ls3=YtF3+1.96*sqrt(diag(Sigma3sg));
ls4=YtF4+1.96*sqrt(diag(Sigma4sg));
ls5=YtF5+1.96*sqrt(diag(Sigma5sg));
ls6=YtF6+1.96*sqrt(diag(Sigma6sg));
ls7=YtF7+1.96*sqrt(diag(Sigma7sg));
ls8=YtF8+1.96*sqrt(diag(Sigma8sg));
ls9=YtF9+1.96*sqrt(diag(Sigma9sg));
ls10=YtF10+1.96*sqrt(diag(Sigma10sg));
ls11=YtF11+1.96*sqrt(diag(Sigma11sg));
ls12=YtF12+1.96*sqrt(diag(Sigma12sg));
ls13=YtF13+1.96*sqrt(diag(Sigma13sg));
ls14=YtF14+1.96*sqrt(diag(Sigma14sg));
ls15=YtF15+1.96*sqrt(diag(Sigma15sg));
ls16=YtF16+1.96*sqrt(diag(Sigma16sg));
ls17=YtF17+1.96*sqrt(diag(Sigma17sg));
ls18=YtF18+1.96*sqrt(diag(Sigma18sg));
ls19=YtF19+1.96*sqrt(diag(Sigma19sg));
ls20=YtF20+1.96*sqrt(diag(Sigma20sg));


li=[...
li1 li2 li3 li4 li5 li6 li7 li8 li9 li10 li11 li12 li13 li14 li15 li16 li17 li18 li19 li20];

ls=[...
ls1 ls2 ls3 ls4 ls5 ls6 ls7 ls8 ls9 ls10 ls11 ls12 ls13 ls14 ls15 ls16 ls17 ls18 ls19 ls20];

Yt=[dccp dfrp dsrp dtrp];%60x4

for i=1:60
        anos(i)=1940+i;
end

for i=1:20
        anosf(i)=2000+i;
end

anos=anos';
anosf=anosf';

figure
plot([anos; anosf],[dccp' YtF(1,:)]','b')
hold on;
plot([anos; anosf],[dccp' li(1,:)]','b:')
hold on;
plot([anos; anosf],[dccp' ls(1,:)]','b:')
hold on;
plot([anos; anosf],[dfrp' YtF(2,:)]','r')
hold on;
```

```
plot([anos; anosf],[dfrp' li(2,:)]','r:')
hold on;
plot([anos; anosf],[dfrp' ls(2,:)]','r:')
hold on;
plot([anos; anosf],[dsrp' YtF(3,:)]','g')
hold on;
plot([anos; anosf],[dsrp' li(3,:)]','g:')
hold on;
plot([anos; anosf],[dsrp' ls(3,:)]','g:')
hold on;
plot([anos; anosf],[dtrp' YtF(4,:)]','m')
hold on;
plot([anos; anosf],[dtrp' li(4,:)]','m:')
hold on;
plot([anos; anosf],[dtrp' ls(4,:)]','m:')
hold on;
for k=1:8
  xx(k)=2000;
  yy(k)=k-2;
end
plot(xx,yy,'k-')
hold on;
AXIS([1940 2020 -1 5])

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%% Pronósticos corregidos

%Generación de matriz MSE para pronósticos corregidos para h=1 hasta 20
Resulta=zeros(4,3*K);
hi=1;
hf=21;% es para el año inmediato anterior. Por ejemplo si se pone hf=21, entonces genera 20 periodos
m=1;
while m<hf
        h=hi;
        Sigmasg=zeros(K,K);
                for i=0:h-1
                        Sigmasg1=f((4*i)+1:4*(i+1),:)*Sigmau*(f((4*i)+1:4*(i+1),:))';
                        Sigmasg=Sigmasg+Sigmasg1;
                end
        Gama=ZZI;
        Gama1=ZZ;
        Omegahf0=zeros(K,K);
        for i=0:h-1
          for j=0:h-1
                Omegahf=trace((((BM')^(h-1-i))*Gama*(BM^(h-1-
j))*Gama1))*f((4*i)+1:4*(i+1),:)*Sigmau*(f((4*j)+1:4*(j+1),:))';
                Omegahf0=Omegahf0+Omegahf;
          end
        end
        Sigmacg=Sigmasg+(1/T)*Omegahf0;
        Resulta1=[Sigmasg Omegahf0 Sigmacg];
        Resulta=[Resulta; Resulta1];
m=m+1;
hi=hi+1;
end
display('                    Sigmasg                              Omegaf0                              Sigmacg')
Resultafin=Resulta(5:hf*4,:)

%Assuming that the data are generated by Gaussian process, we get the following aproximate 95%
%interval forecasts
% Ojo: la estimación puntual del pronóstico NO cambia.
```

```
YtF=[...
YtF1 YtF2 YtF3 YtF4 YtF5 YtF6 YtF7 YtF8 YtF9 YtF10 YtF11 YtF12 YtF13 YtF14 YtF15 YtF16 YtF17 YtF18 YtF19 YtF20
];%4x20, cada matriz es de 4x1

display('          Approximate 95% interval forecast')

[YtF1-1.96*sqrt(diag(Resultafin(1:4,9:12)))    YtF1  YtF1+1.96*sqrt(diag(Resultafin(1:4,9:12)));
 YtF2-1.96*sqrt(diag(Resultafin(5:8,9:12)))    YtF2  YtF2+1.96*sqrt(diag(Resultafin(5:8,9:12)));
 YtF3-1.96*sqrt(diag(Resultafin(9:12,9:12)))    YtF3  YtF3+1.96*sqrt(diag(Resultafin(9:12,9:12)));
 YtF4-1.96*sqrt(diag(Resultafin(13:16,9:12)))  YtF4  YtF4+1.96*sqrt(diag(Resultafin(13:16,9:12)));
 YtF5-1.96*sqrt(diag(Resultafin(17:20,9:12)))  YtF5  YtF5+1.96*sqrt(diag(Resultafin(17:20,9:12)));
 YtF6-1.96*sqrt(diag(Resultafin(21:24,9:12)))  YtF6  YtF6+1.96*sqrt(diag(Resultafin(21:24,9:12)));
 YtF7-1.96*sqrt(diag(Resultafin(25:28,9:12)))  YtF7  YtF7+1.96*sqrt(diag(Resultafin(25:28,9:12)));
 YtF8-1.96*sqrt(diag(Resultafin(29:32,9:12)))  YtF8  YtF8+1.96*sqrt(diag(Resultafin(29:32,9:12)));
 YtF9-1.96*sqrt(diag(Resultafin(33:36,9:12)))  YtF9  YtF9+1.96*sqrt(diag(Resultafin(33:36,9:12)));
YtF10-1.96*sqrt(diag(Resultafin(37:40,9:12)))  YtF10  YtF10+1.96*sqrt(diag(Resultafin(37:40,9:12)));
YtF11-1.96*sqrt(diag(Resultafin(41:44,9:12)))  YtF11 YtF11+1.96*sqrt(diag(Resultafin(41:44,9:12)));
YtF12-1.96*sqrt(diag(Resultafin(45:48,9:12)))  YtF12 YtF12+1.96*sqrt(diag(Resultafin(45:48,9:12)));
YtF13-1.96*sqrt(diag(Resultafin(49:52,9:12)))  YtF13 YtF13+1.96*sqrt(diag(Resultafin(49:52,9:12)));
YtF14-1.96*sqrt(diag(Resultafin(53:56,9:12)))  YtF14 YtF14+1.96*sqrt(diag(Resultafin(53:56,9:12)));
YtF15-1.96*sqrt(diag(Resultafin(57:60,9:12)))  YtF15 YtF15+1.96*sqrt(diag(Resultafin(57:60,9:12)));
YtF16-1.96*sqrt(diag(Resultafin(61:64,9:12)))  YtF16 YtF16+1.96*sqrt(diag(Resultafin(61:64,9:12)));
YtF17-1.96*sqrt(diag(Resultafin(65:68,9:12)))  YtF17 YtF17+1.96*sqrt(diag(Resultafin(65:68,9:12)));
YtF18-1.96*sqrt(diag(Resultafin(69:72,9:12)))  YtF18 YtF18+1.96*sqrt(diag(Resultafin(69:72,9:12)));
YtF19-1.96*sqrt(diag(Resultafin(73:76,9:12)))  YtF19 YtF19+1.96*sqrt(diag(Resultafin(73:76,9:12)));
YtF20-1.96*sqrt(diag(Resultafin(77:80,9:12)))  YtF20 YtF20+1.96*sqrt(diag(Resultafin(77:80,9:12)))
];
% Al centro está la estimación y a los lados el intervalo inferior y el superior

%Generación de límite inferior de pronósticos corregidos

li1c=YtF1-1.96*sqrt(diag(Resultafin(1:4,9:12)));
li2c=YtF2-1.96*sqrt(diag(Resultafin(5:8,9:12)));
li3c=YtF3-1.96*sqrt(diag(Resultafin(9:12,9:12)));
li4c=YtF4-1.96*sqrt(diag(Resultafin(13:16,9:12)));
li5c=YtF5-1.96*sqrt(diag(Resultafin(17:20,9:12)));
li6c=YtF6-1.96*sqrt(diag(Resultafin(21:24,9:12)));
li7c=YtF7-1.96*sqrt(diag(Resultafin(25:28,9:12)));
li8c=YtF8-1.96*sqrt(diag(Resultafin(29:32,9:12)));
li9c=YtF9-1.96*sqrt(diag(Resultafin(33:36,9:12)));
li10c=YtF10-1.96*sqrt(diag(Resultafin(37:40,9:12)));
li11c=YtF11-1.96*sqrt(diag(Resultafin(41:44,9:12)));
li12c=YtF12-1.96*sqrt(diag(Resultafin(45:48,9:12)));
li13c=YtF13-1.96*sqrt(diag(Resultafin(49:52,9:12)));
li14c=YtF14-1.96*sqrt(diag(Resultafin(53:56,9:12)));
li15c=YtF15-1.96*sqrt(diag(Resultafin(57:60,9:12)));
li16c=YtF16-1.96*sqrt(diag(Resultafin(61:64,9:12)));
li17c=YtF17-1.96*sqrt(diag(Resultafin(65:68,9:12)));
li18c=YtF18-1.96*sqrt(diag(Resultafin(69:72,9:12)));
li19c=YtF19-1.96*sqrt(diag(Resultafin(73:76,9:12)));
li20c=YtF20-1.96*sqrt(diag(Resultafin(77:80,9:12)));

%Generación de límite superior de pronósticos corregidos

ls1c=YtF1+1.96*sqrt(diag(Resultafin(1:4,9:12)));
ls2c=YtF2+1.96*sqrt(diag(Resultafin(5:8,9:12)));
ls3c=YtF3+1.96*sqrt(diag(Resultafin(9:12,9:12)));
ls4c=YtF4+1.96*sqrt(diag(Resultafin(13:16,9:12)));
ls5c=YtF5+1.96*sqrt(diag(Resultafin(17:20,9:12)));
ls6c=YtF6+1.96*sqrt(diag(Resultafin(21:24,9:12)));
```

```
ls7c=YtF7+1.96*sqrt(diag(Resultafin(25:28,9:12)));
ls8c=YtF8+1.96*sqrt(diag(Resultafin(29:32,9:12)));
ls9c=YtF9+1.96*sqrt(diag(Resultafin(33:36,9:12)));
ls10c=YtF10+1.96*sqrt(diag(Resultafin(37:40,9:12)));
ls11c=YtF11+1.96*sqrt(diag(Resultafin(41:44,9:12)));
ls12c=YtF12+1.96*sqrt(diag(Resultafin(45:48,9:12)));
ls13c=YtF13+1.96*sqrt(diag(Resultafin(49:52,9:12)));
ls14c=YtF14+1.96*sqrt(diag(Resultafin(53:56,9:12)));
ls15c=YtF15+1.96*sqrt(diag(Resultafin(57:60,9:12)));
ls16c=YtF16+1.96*sqrt(diag(Resultafin(61:64,9:12)));
ls17c=YtF17+1.96*sqrt(diag(Resultafin(65:68,9:12)));
ls18c=YtF18+1.96*sqrt(diag(Resultafin(69:72,9:12)));
ls19c=YtF19+1.96*sqrt(diag(Resultafin(73:76,9:12)));
ls20c=YtF20+1.96*sqrt(diag(Resultafin(77:80,9:12)));

lic=[...
li1c li2c li3c li4c li5c li6c li7c li8c li9c li10c li11c li12c li13c li14c li15c li16c li17c li18c li19c li20c];

lsc=[...
ls1c ls2c ls3c ls4c ls5c ls6c ls7c ls8c ls9c ls10c ls11c ls12c ls13c ls14c ls15c ls16c ls17c ls18c ls19c ls20c];

figure
plot([anos; anosf],[dccp' YtF(1,:)]','b')
hold on;
plot([anos; anosf],[dccp' lic(1,:)]','b:')
hold on;
plot([anos; anosf],[dccp' lsc(1,:)]','b:')
hold on;
plot([anos; anosf],[dfrp' YtF(2,:)]','r')
hold on;
plot([anos; anosf],[dfrp' lic(2,:)]','r:')
hold on;
plot([anos; anosf],[dfrp' lsc(2,:)]','r:')
hold on;
plot([anos; anosf],[dsrp' YtF(3,:)]','g')
hold on;
plot([anos; anosf],[dsrp' lic(3,:)]','g:')
hold on;
plot([anos; anosf],[dsrp' lsc(3,:)]','g:')
hold on;
plot([anos; anosf],[dtrp' YtF(4,:)]','m')
hold on;
plot([anos; anosf],[dtrp' lic(4,:)]','m:')
hold on;
plot([anos; anosf],[dtrp' lsc(4,:)]','m:')
hold on;
for k=1:10
   xx(k)=2000;
   yy(k)=k-3;
end
plot(xx,yy,'k-')
hold on;
AXIS([1940 2020 -1 5])

%Generación de matriz MSE para pronósticos corregidos para h=1 hasta 20

 Sigma1cg=Resultafin(1:4,9:12);
 Sigma2cg=Resultafin(5:8,9:12);
 Sigma3cg=Resultafin(9:12,9:12);
 Sigma4cg=Resultafin(13:16,9:12);
 Sigma5cg=Resultafin(17:20,9:12);
```

```
 Sigma6cg=Resultafin(21:24,9:12);
 Sigma7cg=Resultafin(25:28,9:12);
 Sigma8cg=Resultafin(29:32,9:12);
 Sigma9cg=Resultafin(33:36,9:12);
Sigma10cg=Resultafin(37:40,9:12);
Sigma11cg=Resultafin(41:44,9:12);
Sigma12cg=Resultafin(45:48,9:12);
Sigma13cg=Resultafin(49:52,9:12);
Sigma14cg=Resultafin(53:56,9:12);
Sigma15cg=Resultafin(57:60,9:12);
Sigma16cg=Resultafin(61:64,9:12);
Sigma17cg=Resultafin(65:68,9:12);
Sigma18cg=Resultafin(69:72,9:12);
Sigma19cg=Resultafin(73:76,9:12);
Sigma20cg=Resultafin(77:80,9:12);

Sigmacg2=[...
Sigma1cg; Sigma2cg; Sigma3cg; Sigma4cg; Sigma5cg; Sigma6cg; Sigma7cg; Sigma8cg; Sigma9cg; Sigma10cg; Sigma11cg;
Sigma12cg; Sigma13cg; Sigma14cg; Sigma15cg; Sigma16cg; Sigma17cg; Sigma18cg; Sigma19cg; Sigma20cg];

c44=zeros(4,4);

msecg=[...
Sigma1cg    c44    c44    c44    c44    c44    c44    c44    c44    c44    c44    c44    c44
      c44    c44    c44    c44    c44    c44    c44
c44    Sigma2cg    c44    c44    c44    c44    c44    c44    c44    c44    c44    c44    c44
      c44    c44    c44    c44    c44    c44    c44
c44    c44    Sigma3cg    c44    c44    c44    c44    c44    c44    c44    c44    c44    c44
      c44    c44    c44    c44    c44    c44    c44
c44    c44    c44    Sigma4cg    c44    c44    c44    c44    c44    c44    c44    c44    c44
      c44    c44    c44    c44    c44    c44    c44
c44    c44    c44    c44    Sigma5cg    c44    c44    c44    c44    c44    c44    c44    c44
      c44    c44    c44    c44    c44    c44    c44
c44    c44    c44    c44    c44    Sigma6cg    c44    c44    c44    c44    c44    c44    c44
      c44    c44    c44    c44    c44    c44    c44
c44    c44    c44    c44    c44    c44    Sigma7cg    c44    c44    c44    c44    c44    c44
      c44    c44    c44    c44    c44    c44    c44
c44    c44    c44    c44    c44    c44    c44    Sigma8cg    c44    c44    c44    c44    c44
      c44    c44    c44    c44    c44    c44    c44
c44    c44    c44    c44    c44    c44    c44    c44    Sigma9cg    c44    c44    c44    c44
      c44    c44    c44    c44    c44    c44    c44
c44    c44    c44    c44    c44    c44    c44    c44    c44    Sigma10cg    c44    c44    c44
      c44    c44    c44    c44    c44    c44    c44
c44    c44    c44    c44    c44    c44    c44    c44    c44    c44    Sigma11cg    c44    c44
      c44    c44    c44    c44    c44    c44    c44
c44    c44    c44    c44    c44    c44    c44    c44    c44    c44    c44    Sigma12cg    c44
      c44    c44    c44    c44    c44    c44    c44
c44    c44    c44    c44    c44    c44    c44    c44    c44    c44    c44    c44    Sigma13cg
      c44    c44    c44    c44    c44    c44    c44
c44    c44    c44    c44    c44    c44    c44    c44    c44    c44    c44    c44    c44
      Sigma14cg    c44    c44    c44    c44    c44    c44
c44    c44    c44    c44    c44    c44    c44    c44    c44    c44    c44    c44    c44    c44
      Sigma15cg    c44    c44    c44    c44    c44
c44    c44    c44    c44    c44    c44    c44    c44    c44    c44    c44    c44    c44    c44
      c44    Sigma16cg    c44    c44    c44    c44
c44    c44    c44    c44    c44    c44    c44    c44    c44    c44    c44    c44    c44    c44
      c44    c44    Sigma17cg    c44    c44    c44
c44    c44    c44    c44    c44    c44    c44    c44    c44    c44    c44    c44    c44    c44
      c44    c44    c44    Sigma18cg    c44    c44
c44    c44    c44    c44    c44    c44    c44    c44    c44    c44    c44    c44    c44    c44
      c44    c44    c44    c44    Sigma19cg    c44
```

```
c44	c44	c44	c44	c44	c44	c44	c44	c44	c44	c44	c44	c44	c44
	c44	c44	c44	c44	c44	Sigma20cg
];%80x80

% Esta matriz se usa para ejecutar el programa llamado "General restricted forecastv5".
```

2.5.10 Matlab routine: multiple restricted forecast and compability testing

```
format long g
H=20;
M=13;%número de restricciones linealmente independientes
k=4;

%vector de datos históricos que provienen de las desagregaciones univariadas
%Este vector es un vector apilado y se copia de la hoja de Excel que se llama apilador de datos históricos.
ZH=[...
0.0428571124912705
0.1631646003160400
0.0417424110806452
0.0223859787534498
0.0654452184242305
0.2764242221310350
0.0649035494205284
0.0310814930441202
0.0924562340796012
0.3881081475667230
0.0937216358996681
0.0452494700018725
0.1257241054954320
0.4988647608309230
0.1299340378141720
0.0654042677447738
0.1675272212424930
0.6103695811038690
0.1755059293028130
0.0924600897152863
0.2129537238550700
0.7177248918548650
0.2246682991473800
0.1203773495895030
0.2685513293448160
0.8278150958826570
0.2822681960286570
0.1550145697114990
0.3239587731210360
0.9308282613707530
0.3353801449661800
0.1857598168076660
0.3787719821496940
1.0265946249062400
0.3806649593759050
0.2124363460598100
0.4336748132954200
1.1153412897356300
0.4165553589112000
0.2363796398023300
0.4796977147792790
1.2603878764278100
0.5183258487794150
0.2775703999435420
```

0.5142710568461100
1.3613028966510900
0.5826467430121270
0.3059894144810850
0.5435982190946410
1.4574009925119900
0.6439171803449420
0.3311238496027940
0.5727158682868580
1.5550275069867400
0.7129620699332030
0.3593212142385370
0.5954674444733930
1.6488295630872300
0.7873611841471070
0.3852715126604180
0.6173123813453990
1.7442000054203200
0.8730635818922120
0.4147019936295590
0.6329294358399280
1.8351567689483800
0.9620755094047510
0.4420116944841500
0.6459321138173050
1.9238579430301900
1.0527045874871900
0.4700598046381930
0.6589085780423590
2.0107124740036700
1.1410441087666400
0.5003531469541120
0.6705554500373100
2.0914003809436000
1.2198904873537000
0.5303405349219500
0.6720428982080780
2.2224290123863800
1.3515670711352000
0.5934702951551280
0.6693045111599650
2.3074887764978300
1.4309895091488800
0.6366268794043000
0.6680836723751160
2.3921411575809400
1.5073144371254100
0.6802540524764830
0.6711277656694580
2.4796152239726200
1.5879663894818900
0.7265839965265170
0.6737707816144960
2.5658739197155300
1.6719889081730600
0.7708546431896190
0.6790166615894680
2.6542989874621300
1.7636600559713500
0.8162492658461060
0.6839900935828270

2.7422732591406100
1.8587713263725300
0.8604485232372570
0.6893227257272030
2.8302551375356200
1.9542959792333600
0.9049054959465640
0.6925592360680270
2.9150467516433800
2.0431154584217600
0.9482388552107000
0.6952593184088800
2.9965914803434500
2.1229270698490500
0.9932528447762700
0.7076542259579970
3.0687745038163500
2.2909209208400400
1.0836188628547000
0.7185501324288690
3.1186744661499500
2.3878223275107700
1.1466624984750100
0.7304234815929940
3.1738762546102200
2.4841216440973900
1.2151099698355200
0.7388326981039480
3.2312837598370100
2.5829766997639100
1.2853335699142800
0.7322726742284690
3.2801896165016800
2.6786423814468900
1.3462143249873000
0.7182130028402510
3.3279870362112000
2.7804440783426100
1.4047313525836400
0.6961722272367940
3.3736148886353600
2.8860608219104900
1.4595227879363900
0.6653390520302720
3.4150052115665400
2.9893539847758000
1.5084721085970300
0.6283420317369260
3.4527653273371000
3.0856763794602100
1.5527054057442800
0.5834289267309200
3.4822975378797000
3.1665508054271800
1.5891944552842300
0.5477798206344280
3.4688823789394300
3.1986585967559200
1.6440795653562200
0.5310654444503090
3.4712849893950800

3.2309667920738800
1.6987674645705600
0.5026928935594840
3.4637443578916800
3.2472004383156600
1.7363850835435600
0.4695063275290840
3.4540028393226800
3.2604284494755800
1.7652394345869800
0.4348421702812810
3.4456502894886300
3.2789314303863000
1.7905102442619700
0.4079864221760290
3.4477235818158100
3.3143994724300800
1.8235089535663300
0.3821153790653240
3.4527967742773000
3.3596498881049800
1.8596098827564000
0.3519375868956410
3.4546575742418500
3.4066904104167500
1.8956497509398300
0.3195515377028700
3.4541959912126100
3.4536699907748100
1.9356165396498500
0.2871836473490790
3.4519184137954300
3.4987734137865400
1.9829360272655600
0.2394435132363580
3.4438476128142500
3.5282230521872700
2.0158561327011000
0.2283711892069360
3.4455656753202600
3.5601160700004700
2.0616694380003300
0.2189998086314360
3.4513859956498000
3.5914841165117800
2.1142366455873300
0.2062510095951730
3.4563322822492000
3.6194959768713100
2.1665937467972100
0.1920042202396080
3.4619623786225600
3.6479389984185400
2.2187814878769500
0.1774130172517660
3.4688504084636100
3.6789485856610800
2.2703949653939300
0.1626243130203280
3.4763920689250100
3.7124448282360100

```
2.3204121448603300
0.1477332990097080
3.4837862040044300
3.7471876796037400
2.3681799155343500
0.1321533928864080
3.4893852771880300
3.7806280954364800
2.4127727113322500
0.1533072392344890
3.4915410753403000
3.8105831600908200
2.4538688193173100
];%240x1

%vector de valores futuros (LAS ESTIMACIONES PUNTUALES EN LOS PRONOSTICOS NO CAMBIAN!)
%Se debe generar este vector apilado con los pronósticos corregidos generados del programa anterior
ZF=[...
0.161750947189304
3.496665003287380
3.860136896150660
2.499613147088620
0.159635203034117
3.495838495120080
3.906820295769500
2.540668308626070
0.148367991325296
3.488572436730540
3.946880459073990
2.575260286127080
0.129413567651218
3.475699232895050
3.980496068052100
2.603804608305240
0.104423771458152
3.458079705074970
4.008303357089270
2.627339837961870
0.075129981251118
3.436498157843750
4.030838905361510
2.647050190804980
0.043198853198550
3.411693154904240
4.048555312304970
2.664076832458130
0.010118916907975
3.384362621870250
4.061882070874420
2.679418357919700
-0.022869178512090
3.355148811478380
4.071254477866900
2.693876767277780
-0.054803326623994
3.324620117422050
4.077117440418260
2.708036874098030
-0.085000295117868
3.293257901157380
4.079916674888880
```

```
2.722271107389320
-0.113034599520653
3.261450495542470
4.080085092076300
2.736761899196920
-0.138703734501214
3.229493961667190
4.078028685281590
2.751534529532160
-0.161986218379145
3.197598274561020
4.074114331571360
2.766494536302320
-0.182997515981859
3.165897394952470
4.068660800036750
2.781465216384420
-0.201947592990420
3.134461736485010
4.061933506595690
2.796222100090550
-0.219102679180581
3.103311708048430
4.054143009983620
2.810522461195620
-0.234752810533962
3.072431236583510
4.045446861094350
2.824128887681020
-0.249185894620410
3.041780423257110
4.035954176885170
2.836826680571510
-0.262668399355817
3.011306732541830
4.025732190632930
2.848435384894910];%80x1

%Matrices de coeficientes estimados del modelo VAR
pi1 =[...
1.51593527068       0.162174782139    0.574054627682    0.209032660882
-0.0876411981519 1.50477453653     0.290751238695    0.154752328409
-0.0477359681843 -0.167786556076   1.38687980955     -0.057715681123
0.173299266006    -0.0658436412857 -0.705029940596   1.30708304588]';%4x4

pi2 =[...
-0.619068171897   -0.192136306221   -0.564436153675   -0.261308899307
0.155309197298    -0.451053986565   -0.240609885874   -0.115520936845
-0.0306769499392 0.0988871247145   -0.398732858902   0.0667258776148
-0.154490707764   0.0850405564493   0.688028093304    -0.356638309654]';%4x4

%C        0.0206055860253   0.0390102424871   -0.0362316007769 -0.0143857448846

%Representación VMA
   N1=zeros(4);%4x4
   I1=eye(4);%4x4
 psi0=eye(4);%4x4
 psi1= (psi0*pi1);%4x4
 psi2= (psi1*pi1)+ (psi0*pi2);%4x4
 psi3= (psi2*pi1)+ (psi1*pi2);%4x4
 psi4= (psi3*pi1)+ (psi2*pi2);%4x4
```

```
 psi5= (psi4*pi1)+ (psi3*pi2);%4x4
 psi6= (psi5*pi1)+ (psi4*pi2);%4x4
 psi7= (psi6*pi1)+ (psi5*pi2);%4x4
 psi8= (psi7*pi1)+ (psi6*pi2);%4x4
 psi9= (psi8*pi1)+ (psi7*pi2);%4x4
psi10= (psi9*pi1)+ (psi8*pi2);%4x4
psi11=(psi10*pi1)+ (psi9*pi2);%4x4
psi12=(psi11*pi1)+(psi10*pi2);%4x4
psi13=(psi12*pi1)+(psi11*pi2);%4x4
psi14=(psi13*pi1)+(psi12*pi2);%4x4
psi15=(psi14*pi1)+(psi13*pi2);%4x4
psi16=(psi15*pi1)+(psi14*pi2);%4x4
psi17=(psi16*pi1)+(psi15*pi2);%4x4
psi18=(psi17*pi1)+(psi16*pi2);%4x4
psi19=(psi18*pi1)+(psi17*pi2);%4x4

psi=[...
I1      N1      N1      N1      N1      N1      N1      N1      N1      N1      N1      N1      N1      N1
        N1      N1      N1      N1      N1      N1
psi1    I1      N1      N1      N1      N1      N1      N1      N1      N1      N1      N1      N1      N1
        N1      N1      N1      N1      N1      N1
psi2    psi1    I1      N1      N1      N1      N1      N1      N1      N1      N1      N1      N1      N1
        N1      N1      N1      N1      N1      N1
psi3    psi2    psi1    I1      N1      N1      N1      N1      N1      N1      N1      N1      N1      N1
        N1      N1      N1      N1      N1      N1
psi4    psi3    psi2    psi1    I1      N1      N1      N1      N1      N1      N1      N1      N1      N1
        N1      N1      N1      N1      N1      N1
psi5    psi4    psi3    psi2    psi1    I1      N1      N1      N1      N1      N1      N1      N1      N1
        N1      N1      N1      N1      N1      N1
psi6    psi5    psi4    psi3    psi2    psi1    I1      N1      N1      N1      N1      N1      N1      N1
        N1      N1      N1      N1      N1      N1
psi7    psi6    psi5    psi4    psi3    psi2    psi1    I1      N1      N1      N1      N1      N1      N1
        N1      N1      N1      N1      N1      N1
psi8    psi7    psi6    psi5    psi4    psi3    psi2    psi1    I1      N1      N1      N1      N1      N1
        N1      N1      N1      N1      N1      N1
psi9    psi8    psi7    psi6    psi5    psi4    psi3    psi2    psi1    I1      N1      N1      N1      N1
        N1      N1      N1      N1      N1      N1
psi10   psi9    psi8    psi7    psi6    psi5    psi4    psi3    psi2    psi1    I1      N1      N1      N1
        N1      N1      N1      N1      N1      N1
psi11   psi10   psi9    psi8    psi7    psi6    psi5    psi4    psi3    psi2    psi1    I1      N1      N1
        N1      N1      N1      N1      N1      N1
psi12   psi11   psi10   psi9    psi8    psi7    psi6    psi5    psi4    psi3    psi2    psi1    I1      N1
        N1      N1      N1      N1      N1      N1
psi13   psi12   psi11   psi10   psi9    psi8    psi7    psi6    psi5    psi4    psi3    psi2    psi1    I1
        N1      N1      N1      N1      N1      N1
psi14   psi13   psi12   psi11   psi10   psi9    psi8    psi7    psi6    psi5    psi4    psi3    psi2    psi1
        I1      N1      N1      N1      N1      N1
psi15   psi14   psi13   psi12   psi11   psi10   psi9    psi8    psi7    psi6    psi5    psi4    psi3    psi2
        psi1    I1      N1      N1      N1      N1
psi16   psi15   psi14   psi13   psi12   psi11   psi10   psi9    psi8    psi7    psi6    psi5    psi4    psi3
        psi2    psi1    I1      N1      N1      N1
psi17   psi16   psi15   psi14   psi13   psi12   psi11   psi10   psi9    psi8    psi7    psi6    psi5    psi4
        psi3    psi2    psi1    I1      N1      N1
psi18   psi17   psi16   psi15   psi14   psi13   psi12   psi11   psi10   psi9    psi8    psi7    psi6    psi5
        psi4    psi3    psi2    psi1    I1      N1
psi19   psi18   psi17   psi16   psi15   psi14   psi13   psi12   psi11   psi10   psi9    psi8    psi7    psi6
        psi5    psi4    psi3    psi2    psi1    I1
];%80x80

%Matriz de varianzas-covarianzas contemporáneas de los errores del modelo
Sigmaa=[...
```

```
6.63860345977e-05      -1.28700575957e-06     1.27587663047e-05      2.63052014903e-05
-1.28700575957e-06     0.000162216228908      0.000168990625456      5.66617489479e-05
1.27587663047e-05      0.000168990625456      0.000341721470062      0.000129974549723
2.63052014903e-05      5.66617489479e-05      0.000129974549723      8.59137111394e-05];%4x4


C=[...
0   0   0   0   0   0   0   0   0   1   0   0   0   0
    0   0   0   0   0   0   0   0   0   0   0   0   0
    0   0   0   0   0   0   0   0   0   0   0   0   0
    0   0   0   0   0   0   0   0   0   0   0   0   0
    0   0   0   0   0   0   0   0   0   0   0   0   0
    0   0   0   0   0   0   0   0   0   0   0   0   0
    0
0   0   0   0   0   0   0   0   0   0   1   0   0   0
    0   0   0   0   0   0   0   0   0   0   0   0   0
    0   0   0   0   0   0   0   0   0   0   0   0   0
    0   0   0   0   0   0   0   0   0   0   0   0   0
    0   0   0   0   0   0   0   0   0   0   0   0   0
    0   0   0   0   0   0   0   0   0   0   0   0   0
    0
0   0   0   0   0   0   0   0   0   0   0   1   0   0
    0   0   0   0   0   0   0   0   0   0   0   0   0
    0   0   0   0   0   0   0   0   0   0   0   0   0
    0   0   0   0   0   0   0   0   0   0   0   0   0
    0   0   0   0   0   0   0   0   0   0   0   0   0
    0   0   0   0   0   0   0   0   0   0   0   0   0
    0
0   0   0   0   0   0   0   0   0   0   0   0   0   0
    0   0   0   0   0   0   0   0   1   0   0   0   0
    0   0   0   0   0   0   0   0   0   0   0   0   0
    0   0   0   0   0   0   0   0   0   0   0   0   0
    0   0   0   0   0   0   0   0   0   0   0   0   0
    0   0   0   0   0   0   0   0   0   0   0   0   0
    0
0   0   0   0   0   0   0   0   0   0   0   0   0   0
    0   0   0   0   0   0   0   0   0   1   0   0   0
    0   0   0   0   0   0   0   0   0   0   0   0   0
    0   0   0   0   0   0   0   0   0   0   0   0   0
    0   0   0   0   0   0   0   0   0   0   0   0   0
    0   0   0   0   0   0   0   0   0   0   0   0   0
    0
0   0   0   0   0   0   0   0   0   0   0   0   0   0
    0   0   0   0   0   0   0   0   0   0   0   0   0
    0   0   0   0   0   0   0   0   0   1   0   0   0
    0   0   0   0   0   0   0   0   0   0   0   0   0
    0   0   0   0   0   0   0   0   0   0   0   0   0
    0   0   0   0   0   0   0   0   0   0   0   0   0
    0
0   0   0   0   0   0   0   0   0   0   0   0   0   0
    0   0   0   0   0   0   0   0   0   0   0   0   0
    0   0   0   0   0   0   0   0   0   0   1   0   0
    0   0   0   0   0   0   0   0   0   0   0   0   0
    0   0   0   0   0   0   0   0   0   0   0   0   0
    0   0   0   0   0   0   0   0   0   0   0   0   0
    0
0   0   0   0   0   0   0   0   0   0   0   0   0   0
    0   0   0   0   0   0   0   0   0   0   0   0   0
    0   0   0   0   0   0   0   0   0   0   0   1   0
    0   0   0   0   0   0   0   0   0   0   0   0   0
    0   0   0   0   0   0   0   0   0   0   0   0   0
```

```
0 0 0 0 0 0 0 0 0 0 0 0 0
0
0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 1
0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0
0
0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 1 0 0
0
0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 1 0
0
0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 1
0
0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0
1
];%13x80

sigmau=zeros(13,13);%No hay incertidumbre en las restricciones lineales, según metas de población
IH=eye(20,20);%20x20

%%%
%Matriz de errores promedios de las desagregaciones
Sigmae=[...
4.982290000000031e-005 0 0 0
0 9.511867500000004e-004 0 0
0 0 0.001871007300000 0
0 0 0 9.801173500000012e-004];

msesg=psi*kron(IH,Sigmaa)*psi';%80x80
%msecg=psi*kron(IH,Sigmaa)*psi'+T^-1*Omega; %80x80viene del otro archivo, así que déjese como comentario!

     A= [msecg+Kron(IH,Sigmae)]*C'*inv(C*[msecg+Kron(IH,Sigmae)]*C'+sigmau);
%80x13= 80x13;

%metas de población para la Ciudad Central y cada anillo según datos pronosticados

%opc==1 original dada en el plan de gobierno
%opc==2 propuesta para que las pruebas de compatibilidad no sean significativas al 5%
```

```
opc=2;
if opc==1

Y=[...
3.500541075340300
3.846583160090820
2.525868819317310
3.979880459073990
2.641260286127080
0.124423771458152
3.483079705074970
4.058838905361510
2.679050190804980
0.035196673376006
3.354620117422050
4.127117440418260
2.778036874098030];%13x1

elseif opc==2

Y=[...
3.500541075340300
3.9
2.55
3.979880459073990
2.641260286127080
0.0
3.38
4.058838905361510
2.679050190804980
-0.18
3.0
4.1
2.778036874098030];%13x1

end

  ZFG= ZF  +  A  *[ Y  - C    *ZF];
%80x1= 80x1;

IkH=eye(80,80);%80x80

for i=1941:2020
    ano(i)=i;
end

anos=(ano(:,1941:2020))';


for i=1:60 %ordenadas de datos históricos
        ccpH(i)=(4*i)-3;
        frpH(i)=(4*i)-2;
        srpH(i)=(4*i)-1;
        trpH(i)=4*i;
end

ZHccp=ZH(ccpH);%60x1
ZHfrp=ZH(frpH);%60x1
ZHsrp=ZH(srpH);%60x1
ZHtrp=ZH(trpH);%60x1
```

```
for i=1:20 %ordenadas de pronósticos
        ccpF(i)=(4*i)-3;
        frpF(i)=(4*i)-2;
        srpF(i)=(4*i)-1;
        trpF(i)=4*i;
end

ZFGccp=ZFG(ccpF);%20x1
ZFGfrp=ZFG(frpF);%20x1
ZFGsrp=ZFG(srpF);%20x1
ZFGtrp=ZFG(trpF);%20x1

%%%%%%%%%%%%%%%%%%%%%%Para pronósticos restringidos%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%% Compatibilidad total
Mr=13;
v2r=60-M*2-1;

   d= Y  -  C  *ZF;
%13x1=13x1;

  %K= d' *[inv(C    * [msecg+Kron(IH,Sigmae)] *C')]*d;
  %K= d' *[inv(C    * [msecg] *C')]*d;
   K= d' *[inv(C    * [msecg+Kron(IH,Sigmae)] *C')]*d/Mr;
%1x1=1x1;

SIG=1-fcdf(K,Mr,v2r);%P=fCDF(X,V) returns the F cumulative distribution

% Y es incompatible con C*ZF, si K>F(M)

CompaTotal=[1 K SIG];

disp('                    Prueba Compatibilidad Total'),disp('Contorno      núm. restricción          Kcalc           sig'),disp('Todos'),
disp(CompaTotal);

%%%%%%%%%%%%%%%%%%%%% Compatibilidad parcial
v22r=60-1*2-1;

for m=1:13
   Kp(m,1)=[Y(m,1)-C(m,:)*ZF]'  *[inv(C(m,:)*[msecg+Kron(IH,Sigmae)]*(C(m,:))')]*[Y(m,1)-C(m,:)*ZF];
  %Kp(m,1)=[Y(m,1)-C(m,:)*ZF]'  *[inv(C(m,:)*[msecg               ]*(C(m,:))')]*[Y(m,1)-C(m,:)*ZF];
  Kpi(m)=m;
  sig(m)=1-fcdf(Kp(m,1),1,v22r);
%    1x1=1x1;
end

ccpY=[6,10];
frpY=[1,7,11];
srpY=[2,4,8,12];
trpY=[3,5,9,13];

CompaParcial=[Kpi' Kp sig'];

disp('                    Pruebas Compatibilidad Parcial'),disp('Contorno      núm. restricción          Kpcalc
sig'),disp('frp'),disp(CompaParcial(1,:)),disp('srp'),disp(CompaParcial(2,:)),disp('trp'),disp(CompaParcial(3,:)),disp('srp'),disp(Compa
Parcial(4,:)),disp('trp'),disp(CompaParcial(5,:)),disp('ccp'),disp(CompaParcial(6,:)),disp('frp'),disp(CompaParcial(7,:)),disp('srp'),disp
(CompaParcial(8,:)),disp('trp'),disp(CompaParcial(9,:)),disp('ccp'),disp(CompaParcial(10,:)),disp('frp'),disp(CompaParcial(11,:)),disp
('srp'),disp(CompaParcial(12,:)),disp('trp'),disp(CompaParcial(13,:));

%%%%%%%%%%%%%%%%%%%%% Bandas de predicción

  COVZFG=( A  *   C -IkH)  *[msecg+Kron(IH,Sigmae)]*(  C'*A'   -IkH);
```

```
%  80x80=80x13*13*80-80x80]*[          80x80               ]*(80x13*13x80-80*80);

var=diag(COVZFG);
for i=1:size(var)
        if var(i)<0.00000000000000000001 %20 digitos
                var(i)=0;
        else
                var(i)=var(i);
        end
end
se=var.^(1/2);%80x1

seccpF=se(ccpF);
sefrpF=se(frpF);
sesrpF=se(srpF);
setrpF=se(trpF);

%Bandas de predicción para pronósticos restringidos
%opc2=95 intervalo de confianza al 95%

opc2=95;

if opc2==95
        for i=1:20% puede ser cualquier vector
                Uccp(i)=ZFGccp(i)+1.96*seccpF(i);% 95% nivel de confianza
                Lccp(i)=ZFGccp(i)-1.96*seccpF(i);% 95% nivel de confianza
                Ufrp(i)=ZFGfrp(i)+1.96*sefrpF(i);% 95% nivel de confianza
                Lfrp(i)=ZFGfrp(i)-1.96*sefrpF(i);% 95% nivel de confianza
                Usrp(i)=ZFGsrp(i)+1.96*sesrpF(i);% 95% nivel de confianza
                Lsrp(i)=ZFGsrp(i)-1.96*sesrpF(i);% 95% nivel de confianza
                Utrp(i)=ZFGtrp(i)+1.96*setrpF(i);% 95% nivel de confianza
                Ltrp(i)=ZFGtrp(i)-1.96*setrpF(i);% 95% nivel de confianza
        end
end

 ccpR=[ZHccp;ZFGccp];%80x1
UccpR=[ZHccp; Uccp'];%80x1
LccpR=[ZHccp; Lccp'];%80x1

 frpR=[ZHfrp;ZFGfrp];%80x1
UfrpR=[ZHfrp; Ufrp'];%80x1
LfrpR=[ZHfrp; Lfrp'];%80x1

 srpR=[ZHsrp;ZFGsrp];%80x1
UsrpR=[ZHsrp; Usrp'];%80x1
LsrpR=[ZHsrp; Lsrp'];%80x1

 trpR=[ZHtrp;ZFGtrp];%80x1
UtrpR=[ZHtrp; Utrp'];%80x1
LtrpR=[ZHtrp; Ltrp'];%80x1

Yccp=Y(ccpY);
Yfrp=Y(frpY);
Ysrp=Y(srpY);
Ytrp=Y(trpY);

anosa=[2010 2020]';
anosb=[2003 2010 2020]';
anosc=[2003 2006 2010 2020]';
anosd=[2003 2006 2010 2020]';
```

```
figure;
plot(anos,ccpR,'b')
grid on;
hold on;
plot(anos,UccpR,'b:')
hold on;
plot(anos,LccpR,'b:')
hold on;
plot(anosa,Yccp,'bo')
hold on;
plot(anos,frpR,'r')
hold on;
plot(anos,UfrpR,'r:')
hold on;
plot(anos,LfrpR,'r:')
hold on;
plot(anosb,Yfrp,'ro')
hold on;
plot(anos,srpR,'g')
hold on;
plot(anos,UsrpR,'g:')
hold on;
plot(anos,LsrpR,'g:')
hold on;
plot(anosc,Ysrp,'go')
hold on;
plot(anos,trpR,'m')
hold on;
plot(anos,UtrpR,'m:')
hold on;
plot(anos,LtrpR,'m:')
hold on;
plot(anosd,Ytrp,'mo')
hold on;
for k=1:10
  xx(k)=2000;
  yy(k)=k-3;
end
plot(xx,yy,'r-')
hold on;
legend('ccp','Upper and Lower','bounds of 95% CI','ccp goal','frp','Upper and Lower','bounds of 95% CI','frp goal','srp','Upper and
Lower','bounds of 95% CI','srp goal','trp','Upper and Lower','bounds of 95% CI','trp goal',-1);
xlabel('year')
ylabel('population')
AXIS([1940 2020 -1 5])

for i=2001:2020
   anosal(i)=i;
end

anoout=(anosal(2001:2020))';

disp('                                  Pronósticos Restringidos');
disp('             year          lower CI             ccp             upper CI           std. error');
ccpout=[anoout Lccp' ZFGccp Uccp' seccpF]
disp('             year          lower CI             frp             upper CI           std. error');
frpout=[anoout Lfrp' ZFGfrp Ufrp' sefrpF]
disp('             year          lower CI             srp             upper CI           std. error');
srpout=[anoout Lsrp' ZFGsrp Usrp' sesrpF]
disp('            year          lower CI             trp             upper CI           std. error');
trpout=[anoout Ltrp' ZFGtrp Utrp' setrpF]
```

```
%%%%%%%%%%%%%%%%%%%%%%%Para pronósticos irrestrictos%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%% Datos del conteo 2005
%Se ejecuta para ver compatibilidad de datos del conteo

Y2005=[...
0.146745248535996
3.476569645766740
3.865802399612680
2.694155854509900];%4x1 con estimación INEGI, dejar éste únicamente!

C2005=[...
0    0    0    0    0    0    0    0    0    0    0    0    0    0
     0    0    1    0    0    0    0    0    0    0    0    0    0
     0    0    0    0    0    0    0    0    0    0    0    0    0
     0    0    0    0    0    0    0    0    0    0    0    0    0
     0    0    0    0    0    0    0    0    0    0    0    0    0
     0    0    0    0    0    0    0    0    0    0    0    0    0
     0
0    0    0    0    0    0    0    0    0    0    0    0    0    0
     0    0    0    1    0    0    0    0    0    0    0    0    0
     0    0    0    0    0    0    0    0    0    0    0    0    0
     0    0    0    0    0    0    0    0    0    0    0    0    0
     0    0    0    0    0    0    0    0    0    0    0    0    0
     0    0    0    0    0    0    0    0    0    0    0    0    0
     0
0    0    0    0    0    0    0    0    0    0    0    0    0    0
     0    0    0    0    1    0    0    0    0    0    0    0    0
     0    0    0    0    0    0    0    0    0    0    0    0    0
     0    0    0    0    0    0    0    0    0    0    0    0    0
     0    0    0    0    0    0    0    0    0    0    0    0    0
     0    0    0    0    0    0    0    0    0    0    0    0    0
     0
0    0    0    0    0    0    0    0    0    0    0    0    0    0
     0    0    0    0    0    1    0    0    0    0    0    0    0
     0    0    0    0    0    0    0    0    0    0    0    0    0
     0    0    0    0    0    0    0    0    0    0    0    0    0
     0    0    0    0    0    0    0    0    0    0    0    0    0
     0    0    0    0    0    0    0    0    0    0    0    0    0
     0];%1x80

%%%%%%%%%%%%%%%%%%%% Compatibilidad total
  M2005=4;
  v2=60-M2005*2-1;

  d2005= Y2005  -  C2005  *ZF;%4x1
 %compatibilidad corregida
  K2005= d2005' *[inv(C2005    * [msecg+Kron(IH,Sigmae)] *C2005')]*d2005/M2005;

SIG2005=1-fcdf(K2005,M2005,v2);
CompaTotal2005=[1 K2005 SIG2005];

disp('                         Prueba Compatibilidad Total Datos conteo 2005'),disp('Contorno       núm. restricción          Kcalc
sig'),disp('Todos'), disp(CompaTotal2005);

%%%%%%%%%%%%%%%%%%%% Compatibilidad parcial
   v22=60-1*2-1;

for i=1:4
    K2005(i,1)=[Y2005(i,1)-C2005(i,:)*ZF]'*[inv(C2005(i,:)*[msecg+Kron(IH,Sigmae)]*(C2005(i,:))')]*[Y2005(i,1)-
C2005(i,:)*ZF];
```

```
    K2005pi(i)=i;
    sig2005(i)=1-fcdf(K2005(i,1),1,v22);
end

Compa2005=[K2005pi' K2005 sig2005'];

disp('                    Prueba Compatibilidad Datos conteo 2005'),disp('Contorno       núm. restricción       Kp2005calc
sig'),disp('ccp'),disp(Compa2005(1,:)),disp('frp'),disp(Compa2005(2,:)),disp('srp'),disp(Compa2005(3,:)),disp('trp'),disp(Compa2005(
4,:));

%%%%%%%%%%%%%%%%%%%%%% Bandas de predicción
   COVIRR=msecg+Kron(IH,Sigmae);
%  80x80=[         80x80              ]

varIRR=diag(COVIRR);
for i=1:size(varIRR)
        if varIRR(i)<0.00000000000000000001 %20 digitos
                varIRR(i)=0;
        else
                varIRR(i)=varIRR(i);
        end
end
seIRR=varIRR.^(1/2);%80x1

ZFIRRccp=ZF(ccpF);%20x1
ZFIRRfrp=ZF(frpF);%20x1
ZFIRRsrp=ZF(srpF);%20x1
ZFIRRtrp=ZF(trpF);%20x1

seIRRccpF=seIRR(ccpF);
seIRRfrpF=seIRR(frpF);
seIRRsrpF=seIRR(srpF);
seIRRtrpF=seIRR(trpF);

%Bandas de predicción para pronósticos irrestrictos
%opc3=95 intervalo de confianza al 95%

opc3=95;
if opc3==95
        for i=1:size(seccpF)% puede ser cualquier vector
                UIRRccp(i)=ZFIRRccp(i)+1.96*seIRRccpF(i);% 95% nivel de confianza
                LIRRccp(i)=ZFIRRccp(i)-1.96*seIRRccpF(i);% 95% nivel de confianza
                UIRRfrp(i)=ZFIRRfrp(i)+1.96*seIRRfrpF(i);% 95% nivel de confianza
                LIRRfrp(i)=ZFIRRfrp(i)-1.96*seIRRfrpF(i);% 95% nivel de confianza
                UIRRsrp(i)=ZFIRRsrp(i)+1.96*seIRRsrpF(i);% 95% nivel de confianza
                LIRRsrp(i)=ZFIRRsrp(i)-1.96*seIRRsrpF(i);% 95% nivel de confianza
                UIRRtrp(i)=ZFIRRtrp(i)+1.96*seIRRtrpF(i);% 95% nivel de confianza
                LIRRtrp(i)=ZFIRRtrp(i)-1.96*seIRRtrpF(i);% 95% nivel de confianza
        end
end

 ccpIRR=[ZHccp;ZFIRRccp];%80x1
UccpIRR=[ZHccp; UIRRccp'];%80x1
LccpIRR=[ZHccp; LIRRccp'];%80x1

 frpIRR=[ZHfrp;ZFIRRfrp];%80x1
UfrpIRR=[ZHfrp; UIRRfrp'];%80x1
LfrpIRR=[ZHfrp; LIRRfrp'];%80x1

 srpIRR=[ZHsrp;ZFIRRsrp];%80x1
UsrpIRR=[ZHsrp; UIRRsrp'];%80x1
```

```
LsrpIRR=[ZHsrp; LIRRsrp'];%80x1

 trpIRR=[ZHtrp;ZFIRRtrp];%80x1
UtrpIRR=[ZHtrp; UIRRtrp'];%80x1
LtrpIRR=[ZHtrp; LIRRtrp'];%80x1

Yccp2005=Y2005(1);
Yfrp2005=Y2005(2);
Ysrp2005=Y2005(3);
Ytrp2005=Y2005(4);

anos2005=[2005];

figure;
plot(anos,ccpIRR,'b')
grid on;
hold on;
plot(anos,UccpIRR,'b:')
hold on;
plot(anos,LccpIRR,'b:')
hold on;
plot(anos2005,Yccp2005,'bo')
hold on;
plot(anos,frpIRR,'r')
hold on;
plot(anos,UfrpIRR,'r:')
hold on;
plot(anos,LfrpIRR,'r:')
hold on;
plot(anos2005,Yfrp2005,'ro')
hold on;
plot(anos,srpIRR,'g')
hold on;
plot(anos,UsrpIRR,'g:')
hold on;
plot(anos,LsrpIRR,'g:')
hold on;
plot(anos2005,Ysrp2005,'go')
hold on;
plot(anos,trpIRR,'m')
hold on;
plot(anos,UtrpIRR,'m:')
hold on;
plot(anos,LtrpIRR,'m:')
hold on;
plot(anos2005,Ytrp2005,'mo')
hold on;
%XX=[2000,2000];YY=[2000,14000000];line(XX,YY);
legend('ccp','Upper and Lower','bounds of 95% CI','ccp 2005','frp','Upper and Lower','bounds of 95% CI','frp 2005','srp','Upper and
Lower','bounds of 95% CI','srp 2005','trp','Upper and Lower','bounds of 95% CI','trp 2005',-1);
%legend('ccp','upper and lower','bounds of 95% CI','frp','upper and lower','bounds of 95% CI','srp','upper and lower','bounds of 95%
CI','trp','upper and lower','bounds of 95% CI',-1);
xlabel('year')
ylabel('population')
%AXIS([1940 2020 0 11600000])
for k=1:10
  xx(k)=2000;
  yy(k)=k-3;
end
plot(xx,yy,'r-')
hold on;
```

```
AXIS([1940 2020 -1 5])

disp('                              Pronósticos Irrestrictos');
disp('          year        lower CI         ccp          upper CI        std. error');
ccpout=[anoout LIRRccp' ZFIRRccp UIRRccp' seIRRccpF]
disp('          year        lower CI         frp          upper CI        std. error');
frpout=[anoout LIRRfrp' ZFIRRfrp UIRRfrp' seIRRfrpF]
disp('          year        lower CI         srp          upper CI        std. error');
srpout=[anoout LIRRsrp' ZFIRRsrp UIRRsrp' seIRRsrpF]
disp('          year        lower CI         trp          upper CI        std. error');
trpout=[anoout LIRRtrp' ZFIRRtrp UIRRtrp' seIRRtrpF]

figure;
plot(anos,ccpR,'b')
grid on;
hold on;
plot(anos,UccpR,'b:')
hold on;
plot(anos,LccpR,'b:')
hold on;
plot(anosa,Yccp,'bd')
hold on;
plot(anos,frpR,'r')
hold on;
plot(anos,UfrpR,'r:')
hold on;
plot(anos,LfrpR,'r:')
hold on;
plot(anosb,Yfrp,'r^')
hold on;
plot(anos,srpR,'g')
hold on;
plot(anos,UsrpR,'g:')
hold on;
plot(anos,LsrpR,'g:')
hold on;
plot(anosc,Ysrp,'gs')
hold on;
plot(anos,trpR,'m')
hold on;
plot(anos,UtrpR,'m:')
hold on;
plot(anos,LtrpR,'m:')
hold on;
plot(anosd,Ytrp,'mo')
hold on;
%XX=[2000,2000];YY=[2000,14000000];line(XX,YY);
legend('ccp','Upper and Lower','bounds of 95% CI','ccp goal','frp','Upper and Lower','bounds of 95% CI','frp goal','srp','Upper and Lower','bounds of 95% CI','srp goal','trp','Upper and Lower','bounds of 95% CI','trp goal',-1);
xlabel('year')
ylabel('population')
for k=1:10
  xx(k)=2000;
  yy(k)=k-3;
end
plot(xx,yy,'k-')
hold on;
```

Chapter 3. Smoothing two-dimensional mortality tables with smoothness controlled by the analyst

3.1 Introduction

Smoothed estimates of mortality rates are considered of paramount importance for planning and making strategic decisions in population councils, insurance companies or research centers. There have been several methodological proposals for the analysis, estimation and prediction of mortality rates, *e. g.* the Age-Period-Cohort model (Clayton and Schifflers, 1987), and the Lee and Carter (1992) approach to mortality forecasting as well as other models like that of Brouhns *et al.* (2002). In the two-dimensional context we also found several proposals (e. g. Cleveland and Devlin, 1988) and we emphasize the regression spline approach based on thin plate splines (Dierckx, 1993; De Boor, 2001; Gu and Wahba, 1993; Wood, 2003) or the extension of the B-splines idea (Currie *et al.*, 2004; Eilers and Marx, 1996). However, none of the above proposals considered the possibility of controlling the smoothness achieved by the mortality estimates, to allow for valid comparisons of mortality trends in either the dimension of age and/or the dimension of year. This is the main objective of our work.

When estimating trends in the one-dimensional time-series framework, it is well known (see Gomez, 1999) that the signal extraction method based on the Wiener-Kolmogorov filter, the Kalman filter and Penalized Least Squares provide results equivalent to those produced by the Hodrick-Prescott and the Exponential Smoothing filters employed for economic analysis (King and Rebelo, 1993). On the other hand, Guerrero (2007) showed that Generalized Least Squares (GLS) produces identical results to those already established and he emphasized the fact that the inverse of the corresponding Mean Square Error matrix (MSE) is the sum of two precision matrices. That fact allowed him to use a result to measure the precision share attributable to the smoothness element of the statistical model as did Theil (1963) in a different context. Such a result leads to an index of smoothness that depends only on the smoothing parameter and the sample size of the available data. Therefore, it serves to decide the value of the smoothing parameter as a function of the sample size and some desired smoothness fixed beforehand.

The traditional smoothing approach makes use of the smoothing parameter λ, selected with the aid of Akaike´s Information Criterion (AIC) or with the Bayesian Information Criterion (BIC). Such approach is useful to decide the λ value that optimizes the criterion, but the analyst cannot control the smoothness achieved. Moreover, those automatic criteria are based on a log-likelihood function, so that an underlying statistical model must exist. If we fail to verify the validity of the model, both AIC and BIC become purely numerical criteria, and there are some other drawbacks of using other automatic criteria, like Cross-Validation (Hastie and Tibshirani, 1990). If a dataset is smoothed with a specific λ value, we should realize that a particular amount of smoothness is attained. From a purely descriptive point of view, we suggest to quantify the amount of smoothness with an appropriate measure. We go one step further, since our proposal is to fix in advance a preferred amount of smoothness for all the curves or mortality rates to be smoothed. This idea is in line with that of estimating parameters by way of confidence intervals, where we usually fix the confidence level (say at 95%) to establish valid comparisons. Similarly, we can enhance comparability of smoothed estimates by fixing the percentage of smoothness. Our main argument is that the amount of smoothness can be fixed at the outset to make the smoothed results comparable.

When smoothing data in a univariate framework, it is usually found that the Maximum Likelihood Estimator of the smoothing parameter gets close to zero (see, for instance, Proietti, 2005) implying a trend far from being smooth. In order to avoid that situation, researchers prefer to calibrate the smoothing parameter from a frequency domain perspective. So, they fix that parameter in such a way as to extract meaningful cycles with the Hodrick-Prescott filter. In smoothing splines it is also well known that undersmoothing occurs when the smoothing parameter is estimated by Maximum Likelihood (Kauermann, 2005). Thus, we propose to calibrate the smoothing parameters involved, but we focus this problem from a time domain perspective and emphasize smoothness of the mortality trends. On the other hand, we should also realize that by considering the smoothing constant as a parameter to be estimated from the data at hand we may incur in a statistical cost that reflects itself as inflation in variance. Taylor *et al.* (1996) shows some results related to adding a parameter to specific models. Therefore, since the smoothing parameters are not inherently related with the analysis and interpretation of the data, we prefer to calibrate their values by fixing the smoothness to be attained with them. In summary, our main proposal consists of fixing the smoothing parameters by way of first deciding the amount of smoothness desired for the mortality trend and then applying the usual computational and analytical procedures pertaining to the so-called penalized spline (P-spline). So, we neither provide a new theoretical result nor any new computational procedure for P-splines, but just show how to select the smoothing parameters by deciding beforehand the smoothness to be achieved in an application.

This chapter is organized as follows. In Section 2 we present theoretical results that are already established in the literature for the one-dimensional and the two-dimensional cases. Section 3 is devoted to the study of a two-dimensional smoothness index and the corresponding smoothed estimator of mortality rates. Here some theoretical results that lend support to the use of this index are established. In Section 4 we touch upon some computational aspects. Section 5 illustrates the use of our procedure with applications to data coming from the Continuous Mortality Investigation Bureau of the UK. The numerical examples are useful to appreciate the different log-mortality patterns that can be obtained when each of the marginal smoothness changes, but the joint percentage of smoothness stays fixed. Thus, we should care about the amount of marginal (one-dimensional) as well as the two-dimensional percentages of smoothness achieved in a particular application. In Section 6 we present some final conclusions.

3.2 Some theoretical results

3.2.1 One-dimensional smoothing

Let us consider a B-spline function of degree q and let $\boldsymbol{x} = (x_0, x_1,..., x_n)'$ be a vector of equally spaced knots, so that

$$B_{i,1}(x) = \begin{cases} 1 \text{ for } x \in [x_i, x_{i+1}) \\ 0 \text{ other case} \end{cases} \text{ and } B_{i,q+1}(x) = \frac{x - x_i}{x_{i+q} - x_i} B_{i,q}(x) + \frac{x_{i+q+1} - x}{x_{i+q+1} - x_{i+1}} B_{i+1,q}(x)$$

where $B_{i,q+1}(x) > 0$ for $x_i \le x \le x_{i+k+1}$ and $B_{i,q+1}(x) = 0$ for $x_0 \le x \le x_i$, $x_{i+k+1} \le x \le x_{n+k+1}$ (for details on B-splines, see De Boor, 2001). Now consider a set of B-splines that form a basis and act as predictors in a regression model. Then, given m pairs of observations (x_i, y_i), the objective is to estimate the regression of y on x by Least Squares. The P-spline method reduces the number of splines by placing a smoothness penalty on the differences of coefficients for adjacent B-splines (Eilers and Marx, 1996). Then, it seeks to minimize

$$S = \sum_{i=1}^{m} (y_i - \hat{y}_{(\alpha)_i})^2 + \lambda \sum_{j=3}^{p} (\Delta^2 \alpha_j)^2 \tag{2.1}$$

with $\hat{y}_{(\alpha)_i} = \sum_{j=1}^{p} \alpha_j B_j(x_i, q)$, where p is the number of B-splines, the α_j s are constant coefficients and the $B_j(x,q)$ form a basis of degree q. We use cubic splines (q = 3) to offset the cost of computer operations and the potential of shaping the resulting B-splines. The parameter λ trades off fit against smoothness induced by the second order difference $\Delta^2 \alpha_j = \alpha_j - 2\alpha_{j-1} + \alpha_{j-2}$.

Let B be a matrix with elements $B_{ij} = B_j(x_i)$, so that its columns are B-splines with local support evaluated at the different values of x. Let $\Omega = D_2{}'D_2$ be a symmetric matrix, where D_2 is the $(m-2)\times m$ matrix representation of the difference operator Δ^2, that is

$$D_2 = \begin{pmatrix} 1 & -2 & 1 & 0 & 0 & \dots & 0 & 0 & 0 & 0 \\ 0 & 1 & -2 & 1 & 0 & & 0 & 0 & 0 & 0 \\ & & & & & \dots & & & & \\ 0 & 0 & 0 & 0 & 0 & \dots & 0 & 1 & -2 & 1 \end{pmatrix}.$$

For ease of exposition, let $p = m$, so that equation (2.1) can be expressed in matrix notation as $S = (\boldsymbol{y} - B\boldsymbol{\alpha})'(\boldsymbol{y} - B\boldsymbol{\alpha}) + \lambda\boldsymbol{\alpha}'\Omega\boldsymbol{\alpha}$. If we minimize this expression with respect to $\boldsymbol{\alpha}$, we get the following result (see Eilers and Marx 1996, eq. 13), that is

$$B'\boldsymbol{y} = (B'B + \lambda\Omega)\hat{\boldsymbol{\alpha}}. \tag{2.2}$$

Moreover, the solution vector $\hat{\boldsymbol{y}}_{(\alpha)}$ can be written in terms of an $m\times m$ nonsingular natural-spline basis N, as in Hastie and Tibshirani (1990, pp. 28-29). Now, let $\hat{\boldsymbol{\beta}}$ be the transformed version of $\hat{\boldsymbol{\alpha}}$ corresponding to this change of basis. Then, we get

$$\hat{\boldsymbol{y}}_{(\alpha)} = N\hat{\boldsymbol{\beta}} = N(N'N + \boldsymbol{\lambda}\Omega)^{-1}N'\boldsymbol{y} = (I_m + \lambda K'K)^{-1}\,\boldsymbol{y} \tag{2.3}$$

where $K = D_2N^{-1}$. The matrix $\lambda K'K$ has two eigenvalues equal to zero and $m-2$ greater than zero, so that it is positive semidefinite and $I_m + \lambda K'K$ is positive definite.

From a different perspective, we now write the model as

$$\boldsymbol{y} = \boldsymbol{y}_{(\alpha)} + \boldsymbol{\eta} \tag{2.4}$$

with $\mathrm{E}(\boldsymbol{\eta}) = \boldsymbol{0}$ and $\mathrm{Var}(\boldsymbol{\eta}) = \sigma_\eta^2 I_m$, where $\boldsymbol{y}_{(\alpha)} = \sum_{j=1}^{m}\alpha_j N_j(x,q)$. To induce smoothness, let

$$D_2N^{-1}\boldsymbol{y}_{(\alpha)} = \boldsymbol{\varepsilon} \tag{2.5}$$

with $\mathrm{E}(\boldsymbol{\varepsilon})=\boldsymbol{0}$ and $\mathrm{Var}(\boldsymbol{\varepsilon})=\sigma_\varepsilon^2 I_{m-2}$. Since $\mathrm{E}(\boldsymbol{\varepsilon\eta'})=0$, we have

$$\begin{pmatrix}\boldsymbol{y}\\ \boldsymbol{0}\end{pmatrix}=\begin{pmatrix}I_m\\ D_2N^{-1}\end{pmatrix}\boldsymbol{y}_{(\alpha)}+\begin{pmatrix}\boldsymbol{\eta}\\ -\varepsilon\end{pmatrix}\ \text{ with } \mathrm{E}\begin{pmatrix}\boldsymbol{\eta}\\ -\varepsilon\end{pmatrix}=\boldsymbol{0}\ \text{ and } \mathrm{Var}\begin{pmatrix}\boldsymbol{\eta}\\ -\varepsilon\end{pmatrix}=\begin{pmatrix}\sigma_\eta^2 I_m & 0\\ 0 & \sigma_\varepsilon^2 I_{m-2}\end{pmatrix}=\Sigma .$$

If we make $\boldsymbol{Y}=\begin{pmatrix}\boldsymbol{y}\\ \boldsymbol{0}\end{pmatrix}$, $A=\begin{pmatrix}I_m\\ D_2N^{-1}\end{pmatrix}$, $\boldsymbol{X}=\boldsymbol{y}_{(\alpha)}$ and $\boldsymbol{E}=\begin{pmatrix}\boldsymbol{\eta}\\ -\boldsymbol{\varepsilon}\end{pmatrix}$, we can write $\boldsymbol{Y}=A\boldsymbol{X}+\boldsymbol{E}$. Then, the GLS solution is given by $\hat{\boldsymbol{X}}=\left(A'\Sigma^{-1}A\right)^{-1}A'\Sigma^{-1}Y$, that is

$$\hat{\boldsymbol{y}}_{(\alpha)}=(I_m+(\sigma_\varepsilon^{-2}/\sigma_\eta^{-2})K'K)^{-1}\boldsymbol{y} \tag{2.6}$$

and its corresponding MSE matrix is

$$\Gamma=\mathrm{Var}(\hat{\boldsymbol{y}}_{(\alpha)})=(\sigma_\eta^{-2}I_m+\sigma_\varepsilon^{-2}K'K)^{-1}. \tag{2.7}$$

Thus, if we let $\lambda=\sigma_\eta^2/\sigma_\varepsilon^2$ the two approaches produce the same $\hat{\boldsymbol{y}}_{(\alpha)}$.

3.2.2 Two-dimensional smoothing

In a mortality table, let the vectors $\boldsymbol{d}=(d_{11},\ldots,d_{m1},\ldots,d_{1n},\ldots,d_{mn})'$, $\boldsymbol{\mu}=(\mu_{11},\ldots,\mu_{m1},\ldots,\mu_{1n},\ldots,\mu_{mn})'$ and $\boldsymbol{e}=(e_{11},\ldots,e_{m1},\ldots,e_{1n},\ldots,e_{mn})'$ denote deaths occurred, forces of mortality and exposures to the risk of dying for ages 1 to m, and years 1 to n, respectively. We assume that d_{in} is a realization of a Poisson process with mean $e_{ij}\mu_{ij}$, for $i=1,\ldots,m$ and $j=1,\ldots,n$. In what follows we consider a smoothed estimate of the vector $\boldsymbol{Y}=(Y_{11},\ldots,Y_{m1},\ldots,Y_{1n},\ldots,Y_{mn})'$, where $Y_{ij}=\log(d_{ij}/e_{ij})$ is the crude force of mortality, for age $i=1,\ldots,m$ and year $j=1,\ldots,n$. Let $B_a=B(x_a)$ be an $m\times m_a$ regression matrix of B-splines based on the explanatory variable x_a for ages. Similarly, let $B_y=B(x_y)$ be an $n\times n_y$ regression matrix of B-splines based on x_y for years. The $mn\times m_a n_y$ regression matrix B is given by the Kronecker product $B = B_y \otimes B_a$. This matrix is associated to an mn vector $\boldsymbol{\alpha}$ of regression coefficients, to be estimated.

For the one-dimensional case, the penalty function is $\boldsymbol{\alpha}' D_2' D_2 \boldsymbol{\alpha}$ (Eilers and Marx, 1996). For the two-dimensional case, that idea is extended using in the age dimension the smoothness restriction

$$\boldsymbol{\alpha}'(I_n \otimes D_a' D_a)\boldsymbol{\alpha}, \tag{2.8}$$

while in the dimension of years we have

$$\boldsymbol{\alpha}'(D_y' D_y \otimes I_m)\boldsymbol{\alpha}. \tag{2.9}$$

Thus, D_a and D_y are difference matrices on columns (age) and rows (year) respectively. Now, let B be a matrix with $n_y = n$ and $m_a = m$, so that the problem can be expressed as $\min_{\boldsymbol{\alpha}} [(\boldsymbol{Y}\text{-}B\alpha)' (\boldsymbol{Y}\text{-}B\alpha) + \lambda_a \boldsymbol{\alpha}'(I_n \otimes D_a' D_a)\boldsymbol{\alpha} + \lambda_y \boldsymbol{\alpha}'(D_y' D_y \otimes I_m)\boldsymbol{\alpha}]$ and its solution is the estimating equation

$$B'\boldsymbol{Y} = [B'B + \lambda_a(I_n \otimes D_a' D_a) + \lambda_y(D_y' D_y \otimes I_m)]\,\hat{\boldsymbol{\alpha}}. \tag{2.10}$$

As with the one-dimensional case, for theoretical purposes we shall employ two nonsingular $m\times m$ and $n\times n$ natural-spline bases, N_a for age and N_y for year. Hence the solution will use the $mn \times mn$ nonsingular natural-spline basis $N_{ay} = N_y \otimes N_a$ and $\hat{\boldsymbol{\gamma}}$ (the transformed version of $\hat{\boldsymbol{\alpha}}$ corresponding to the new basis). Thus, we get $\hat{\boldsymbol{Y}}_{(\alpha)} = N_{ay}\,\hat{\boldsymbol{\gamma}}$, *i.e.*

$$\begin{aligned}\hat{\boldsymbol{Y}}_{(\alpha)} &= N_{ay}\left[N_{ay}' N_{ay} + \lambda_a(I_n \otimes D_a' D_a) + \lambda_y(D_y' D_y \otimes I_m)\right]^{-1} N_{ay}'\boldsymbol{Y} \\ &= (I_{mn} + \lambda_a K_a' K_a + \lambda_y K_y' K_y)^{-1}\boldsymbol{Y}\end{aligned} \tag{2.11}$$

with $K_a = (I_n \otimes D_a)N_{ay}^{-1}$ and $K_y = (D_y \otimes I_m)N_{ay}^{-1}$. The matrix $I_{mn} + \lambda_a K_a' K_a + \lambda_y K_y' K_y$ is positive definite since all its eigenvalues are non-negative.

From the second perspective, let us consider the mn stacked vector $\boldsymbol{Y} = (Y_{11},...,Y_{m1},...,Y_{1n},...,Y_{mn})'$ whose elements are crude forces of mortality in logs, that is $Y_{ij} = \log(d_{ij}/e_{ij})$ for i=1,…, m and j=1,…, n. We now use the model

$$\boldsymbol{Y} = \boldsymbol{Y}_{(\alpha)} + \boldsymbol{\Psi} \tag{2.12}$$

with $\mathrm{E}(\boldsymbol{\Psi}) = \boldsymbol{0}$ and $\mathrm{Var}(\boldsymbol{\Psi}) = \sigma^2_{\Psi} I_{mn}$, where $Y_{ij(\alpha)} = \sum_{j=1}^{n}\sum_{k=1}^{m} \alpha_{jk} N_{ay,jk}(\mathrm{x}_1, \mathrm{x}_2, q)$. To induce smoothness in the dimension of age, we assume that

$$K_a\, \boldsymbol{Y}_{(\alpha)} = \boldsymbol{\Theta} \tag{2.13}$$

with $\mathrm{E}(\boldsymbol{\Theta}) = \boldsymbol{0}$ and $\mathrm{Var}(\boldsymbol{\Theta}) = \sigma^2_{\Theta} I_{(m-2)n}$. Similarly, to induce smoothness in years, let

$$K_y\, \boldsymbol{Y}_{(\alpha)} = \boldsymbol{\Phi} \tag{2.14}$$

with $\mathrm{E}(\boldsymbol{\Phi}) = \boldsymbol{0}$ and $\mathrm{Var}(\boldsymbol{\Phi}) = \sigma^2_{\Phi} I_{(n-2)m}$. Since $\mathrm{E}(\boldsymbol{\Theta\Psi'}) = 0$ and $\mathrm{E}(\boldsymbol{\Phi\Psi'}) = 0$, we have

$$\begin{pmatrix} \boldsymbol{Y} \\ \boldsymbol{0} \\ \boldsymbol{0} \end{pmatrix} = \begin{pmatrix} I_{mn} \\ K_a \\ K_y \end{pmatrix} \boldsymbol{Y}_{(\alpha)} + \begin{pmatrix} \boldsymbol{\Psi} \\ -\boldsymbol{\Theta} \\ -\boldsymbol{\Phi} \end{pmatrix} \text{ with } \mathrm{E}\begin{pmatrix} \boldsymbol{\Psi} \\ -\boldsymbol{\Theta} \\ -\boldsymbol{\Phi} \end{pmatrix} = \boldsymbol{0} \text{ and } \mathrm{Var}\begin{pmatrix} \boldsymbol{\Psi} \\ -\boldsymbol{\Theta} \\ -\boldsymbol{\Phi} \end{pmatrix} = \begin{pmatrix} \sigma^2_{\Psi} I_{mn} & 0 & 0 \\ 0 & \sigma^2_{\Theta} I_{(m-2)n} & 0 \\ 0 & 0 & \sigma^2_{\Phi} I_{(n-2)m} \end{pmatrix}$$

Then, the GLS solution produces the same solution as (3.4), that is

$$\hat{\boldsymbol{Y}}_{(\alpha)} = (I_{mn} + \lambda_a K'_a K_a + \lambda_y K'_y K_y)^{-1} \boldsymbol{Y} \tag{2.15}$$

with $\lambda_a = \sigma^2_{\Psi}/\sigma^2_{\Theta}$ and $\lambda_y = \sigma^2_{\Psi}/\sigma^2_{\Phi}$. The resulting MSE matrix is given by

$$\Gamma = \mathrm{Var}(\hat{\boldsymbol{Y}}_{(\alpha)}) = (\sigma^{-2}_{\Psi} I_{mn} + \sigma^{-2}_{\Theta} K'_a K_a + \sigma^{-2}_{\Phi} K'_y K_y)^{-1}. \tag{2.16}$$

Then the results are the same using both approaches. In general, to perform numerical calculations it is not necessary to estimate the inverse matrices (2.6) and (2.7) or (3.8) and (3.9) (since their sizes could be too large), but rather we use an empirical rule (Ruppert, 2002), whose objective is to use a number of B-splines much smaller than the number of observations.

3.3 Smoothness indices and their use to choose the smoothing parameters

The inverse matrix Γ^{-1} is the sum of three precision matrices, $\sigma_{\Psi}^{-2} I_{mn}$, $\sigma_{\Theta}^{-2} K'_a K_a$ and $\sigma_{\Phi}^{-2} K'_y K_y$ associated with models (2.12), (2.13) and (2.14), respectively. Thus, we follow the idea of using an index to measure the proportion of P in $(P+Q)^{-1}$, where P and Q are $n \times n$ positive definite matrices (Guerrero, 2008). The appropriate index is given by

$$\Lambda(P; P+Q) = tr[P(P+Q)^{-1}]/n \tag{3.1}$$

where $tr(.)$ denotes trace of a matrix. When P and Q are precision matrices, this index measures relative precision and has the following properties: (i) it satisfies an adding-up criterion, in the sense that $\Lambda(P; P+Q) + \Lambda(Q; P+Q) = 1$; (ii) it takes on values between zero and one; (iii) it is invariant under linear nonsingular transformations of the variable involved; and (iv) it behaves linearly. Properties (i), (ii) and (iii) are necessary conditions for obtaining the measure, and property (iv) ensures its uniqueness (Theil, 1963).

We now extend (3.1) to produce a measure applicable to the case of three matrices. Such a measure will allow us to define a smoothness index related to ages and years. That is, we propose a function $\Lambda(P; P+Q_a+Q_y)$ to measure the proportion of P in $(P+Q_a+Q_y)^{-1}$ and similarly, a measure of the proportion corresponding to Q_a or Q_y in $(P+Q_a+Q_y)^{-1}$. Thus, we establish the following result.

Proposition 1. Let P, Q_a and Q_y be three symmetric, positive definite or semidefinite matrices. A scalar index that measures the proportion of P in $(P+Q_a+Q_y)^{-1}$ is given by

$$\Lambda(P; P+Q_a+Q_y) = tr\left[P\,(P+Q_a+Q_y)^{-1}\right]/mn \tag{3.2}$$

This measure satisfies the aforementioned criteria: adding-up, zero-unit, invariance under linear nonsingular transformations and linearity.

Proof. See Appendix 3.7.1.

We can use (3.2) to quantify the proportion of precision attributable to the use of (2.14) and (2.15) in the precision matrix $\Gamma^{-1} = P + Q_a + Q_y$ with $P = \sigma_{\Psi}^{-2} I_{mn}$, $Q_a = \sigma_{\Theta}^{-2} K'_a K_a$ and $Q_y = \sigma_{\Phi}^{-2} K'_y K_y$. To that end, we propose the following two-dimensional indices of smoothness: attributable to age,

$$\mathrm{S}_{a\bullet}(\lambda_a,\ \lambda_y;\ m,\ n) = \Lambda(Q_a;\Gamma^{-1}) = tr[\lambda_a K'_a K_a (I_{mn} + \lambda_a K'_a K_a + \lambda_y K'_y K_y)^{-1}]/mn. \tag{3.3}$$

and attributable to year (with respect to the total in both cases)

$$\mathrm{S}_{\bullet y}(\lambda_a,\ \lambda_y;\ m,\ n) = \Lambda(Q_y;\Gamma^{-1}) = tr[\lambda_y K'_y K_y (I_{mn} + \lambda_a K'_a K_a + \lambda_y K'_y K_y)^{-1}]/mn. \tag{3.4}$$

Marginal indices of smoothness can also be obtained from these expressions. Indeed, if we make $n = 1$ in (3.3), which implies to consider just a single year and assume $\lambda_y = 0$, so that no smoothness is attributable to years, the marginal smoothness attributable to age is given by $\mathrm{S}_a(\lambda_a;\ m) = \mathrm{S}_{a\bullet}(\lambda_a, 0;\ m,\ 1) = 1 - tr[(I_m + \lambda_a K'_a K_a)^{-1}]/m$, where $K_a = D_a N_a^{-1}$ is an $(m-2)\times m$ matrix. Similarly, if we make $m = 1$ and $\lambda_a = 0$ in (3.4), the marginal smoothness attributable to years is $\mathrm{S}_y(\lambda_y;\ n) = \mathrm{S}_{\bullet y}(0, \lambda_y;\ 1,\ n) = 1 - tr[(I_n + \lambda_y K'_y K_y)^{-1}]/n$, where $K_y = D_y N_y^{-1}$ is an $(n-2)\times n$ matrix. In a mortality table, it is convenient to smooth jointly, with respect to both ages and years. Therefore, in agreement with (3.3) and (3.4), the two-dimensional index is given by

$$\mathrm{S}_{ay}(\lambda_a, \lambda_y;\ m,\ n) = 1 - tr[(I_{mn} + \lambda_a K'_a K_a + \lambda_y K'_y K_y)^{-1}]/mn \tag{3.5}$$

and we obtain the following result $\mathrm{S}_{ay}(\lambda_a, \lambda_y;\ m,\ n) = \mathrm{S}_{a\bullet}(\lambda_a,\ \lambda_y;\ m,\ n) + \mathrm{S}_{\bullet y}(\lambda_a,\ \lambda_y;\ m,\ n)$.

3.3.1 Choosing the smoothing parameters to achieve a desired percentage of smoothness

The index $S_{ay}(\lambda_a, \lambda_y; m, n)$ can be expressed in percentage terms by writing it as $100\,S_{ay}(\lambda_a, \lambda_y; m, n)\%$ or just as $S_{ay}\%$, in this form we interpret it as percentage of smoothness achieved. Figure 3.1 shows the joint smoothness index for *m* = 30 ages and *n* = 30 years, when varying the smoothing parameters.

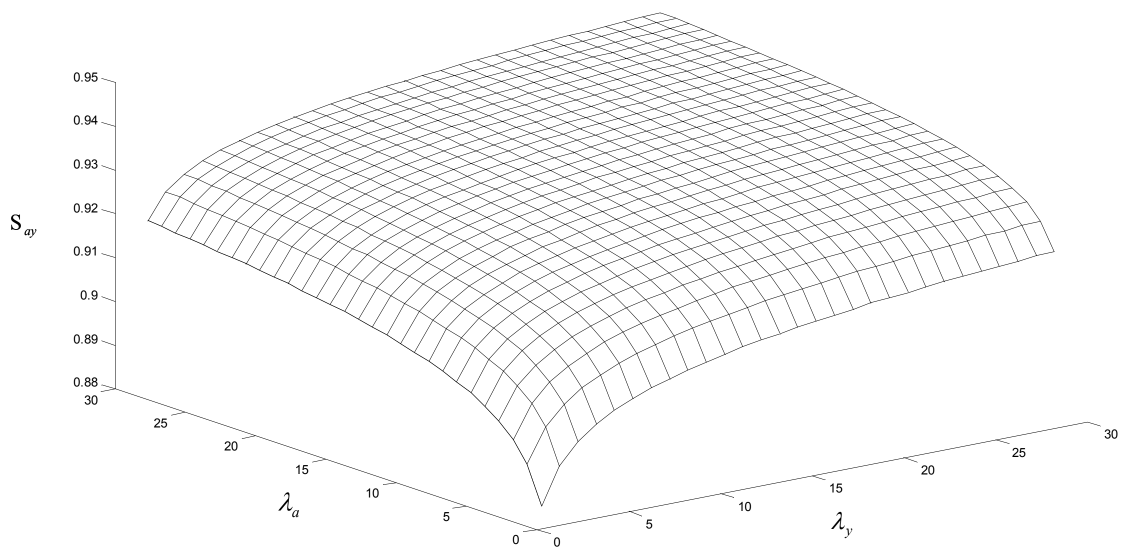


Figure 3.1. Two-dimensional index of smoothness $S_{ay}(\lambda_a, \lambda_y; m = 30, n = 30)$ for $\lambda_a, \lambda_y = 1,\ldots, 29$

The index of smoothness depends only on the smoothing parameters λ_a and λ_y, as well as on *m* and *n*, since K_a and K_y are fully determined by *m* and *n*. Thus, in order to achieve a joint percentage of smoothness for a given dataset (*m* and *n* fixed) we just need to decide the value of $S_{ay}\%$ and solve for the corresponding λ values. Once the smoothing parameters are found we can get the smoother matrix, whose trace is known as *effective dimension* or *degrees of freedom* (*df*) of the model (Hastie and Tibshirani, 1990). That is, *df* = $tr[(I_m + \lambda K'K)^{-1}]$ in the one-dimensional case and *df* = $tr[(I_{mn} + \lambda_a K_a'K_a + \lambda_y K_y'K_y)^{-1}]$ in the two-dimensional one. Since our index

$S_{ay}(\lambda_a, \lambda_y; m, n)$ is a linear transformation of that trace, it can be considered a reparameterization of the *df* of the model. Then, our proposal is in line with Hastie and Tibshirani's comment (1990, p. 52) "it is reasonable to select the value of a smoothing parameter simply by specifying the *df* of the smooth." Some advantages of using our reparameterization of the *df*, besides its sensible interpretation, are the properties of the smoothness index and the smoothed estimate established in propositions 1 through 4.

3.3.2 Additional properties of the smoothness index

In the one-dimensional case, it is well-known that $tr[(I_m + \lambda K'K)^{-1}] \to d$ as $\lambda \to \infty$, where $d = 2$ is the degree of the difference operator (see Eilers and Marx, 1996, p. 94). Therefore, the smoothness to be achieved with *m* observations is such that $S(\lambda; m) \to 1 - 2/m$ as $\lambda \to \infty$. Correspondingly, for the two-dimensional index of smoothness we deduce the following bounds.

Proposition 2. Let the two-dimensional index of smoothness be given by (3.5) and let $\gamma_{a,i}$ and $\gamma_{y,j}$ be the eigenvalues of $K'_a K_a$ and $K'_y K_y$, respectively. Then, if we interpret $\lambda_a, \lambda_y = 0$ as $\lambda_a, \lambda_y \to 0$ and $\lambda_a, \lambda_y = \infty$ as $\lambda_a, \lambda_y \to \infty$, we have: i) $S_{ay}(0, 0; m, n) \to 0$; ii) $S_{ay}(0, \lambda_y; m, n) \to 1-\left(\sum_{j=1}^{m-2}\frac{1}{1+\lambda_y\gamma_{y,j}}+2\right)/m$; iii) $S_{ay}(0, \infty; m, n) \to 1-2/n$; iv) $S_{ay}(\lambda_a, 0; m, n) \to 1-\left(\sum_{i=1}^{n-2}\frac{1}{1+\lambda_a\gamma_{a,i}}+2\right)/n$; v) $S_{ay}(\lambda_a, \infty; m, n) \to 1 - \frac{2}{mn}\left(\sum_{i=1}^{n-2}\frac{1}{1+\lambda_a\gamma_{a,i}}+2\right)$; vi) $S_{ay}(\infty, 0; m, n) \to 1 - 2/m$; vii) $S_{ay}(\infty, \lambda_y; m, n) \to 1 - \frac{2}{mn}\left(\sum_{j=1}^{m-2}\frac{1}{1+\lambda_y\gamma_{y,j}}+2\right)$; and viii) $S_{ay}(\infty, \infty; m, n) \to 1 - 4/mn$.

Proof. See Appendix 3.7.2.

To appreciate the relationship between the two-dimensional and the one-dimensional indexes of smoothness, we collapse one of the two dimensions of the mortality table. In what follows we consider ages specifically, but the argument is valid for years too. We first recall expression (2.3), written in two-dimensional notation as the $m \times 1$ vector of mortality rates smoothed marginally by ages, $\hat{\boldsymbol{y}}_{(\alpha)} = (I_m + \lambda_a K'K)^{-1} \boldsymbol{y}$. Then, if we make $N_y = I_n$, $N_a^{-1} = N^{-1}$ and $D_a = D_2$ in $\hat{\boldsymbol{Y}}_{(\alpha)} = I_n \otimes I_m + \lambda_a (N_y \otimes N_a)^{-1}{}'(I_n \otimes D_a)'(I_n \otimes D_a)(N_y \otimes N_a)^{-1}]^{-1} \boldsymbol{Y}$, we get the following.

Proposition 3. Let the $mn \times 1$ two-dimensional smoothed estimator for ages be given by $\hat{\boldsymbol{Y}}_{(\alpha)} = (I_{mn} + \lambda_a K_a{}'K_a)^{-1} \boldsymbol{Y}$. This estimator employs the same smoother matrix and parameter λ_a used by the one-dimensional estimator, since it can be expressed as

$$\hat{\boldsymbol{Y}}_{(\alpha)} = [\, I_n \otimes (I_m + \lambda_a K'K)^{-1} \,] \; \boldsymbol{Y} \,. \tag{3.6}$$

The following proposition relates the two-dimensional index of smoothness to the one-dimensional ones. This result is useful to know how much smoothness can be achieved in two dimensions, when marginal amounts of smoothness attributable to years or ages are known.

Proposition 4. The two-dimensional index of smoothness can be expressed in terms of the one-dimensional indices of smoothness for years and ages as follows

$$\begin{aligned} S_{ay}(\lambda_a, \lambda_y; m, n) = [S_a(\lambda_a; m) + S_y(\lambda_y; n)]/2 + tr\{[\lambda_y K_y^{'} K_y (I_{mn} + \lambda_a K_a^{'} K_a)^{-1} \\ + \lambda_a K_a^{'} K_a (I_{mn} + \lambda_y K_y^{'} K_y)^{-1}](I_{mn} + \lambda_a K_a^{'} K_a + \lambda_y K_y^{'} K_y)^{-1}\}/2mn \,. \end{aligned} \tag{3.7}$$

Proof. See Appendix 3.7.4.

We see that the two-dimensional index of smoothness is made up of an average of the two one-dimensional indices, plus an element of interaction. Some particular cases of Proposition 4 are the following: i) $S_{ay}(0, \lambda_y;\ m,\ n) = [S_y(\lambda_y;\ n) + S_{\bullet y}(0, \lambda_y;\ m,\ n)]\ /2$ and ii) $S_{ay}(\lambda_a,\ 0;\ m,\ n) = [S_a(\lambda_a;\ m) + S_{a\bullet}(\lambda_a,\ 0;\ m,\ n)]\ /2$. Thus, if we choose not to smooth in either the age or the year dimension, the two-dimensional index becomes an average of the marginal and the two-dimensional indices of smoothness for the dimension where smoothness is sought.

3.4 Computational aspects

It is well known that smoothing log mortality by (Penalized) Least Squares is sub-optimal and Penalized Maximum Likelihood is better, as it is shown in Pawitan (2001). However, for computations we decided to follow the procedure suggested by Currie *et al.* (2004) in order to establish comparisons with our proposal. They used a Generalized Linear Model (GLM) and its penalized likelihood function, from which they derived the following system of equations $B'(y - \mu) = P\boldsymbol{a}$. This system is solved with a penalized version of the *Scoring* algorithm, that is

$$(B'\tilde{W}B + P)\boldsymbol{a} = B'\tilde{W}B\tilde{\boldsymbol{a}} + B'(y - \tilde{\mu}) \tag{4.1}$$

where B is the B-splines regression matrix, $P = \lambda D'_2 D_2$ is the penalty matrix, $\tilde{\boldsymbol{a}}$ and $\tilde{\mu}$ are approximations to the solution, and $\tilde{W}$ is a diagonal matrix with weights $w_{ii}^{-1} = (\partial \mu_i / \partial \eta_i)^2 / v_i$, where v_i is the variance of y_i given its mean μ_i, and $\eta_i = \sum_{j=1}^{p} \alpha_j B_j(x_i, q)$. For Poisson errors, which is the usual assumption for mortality rates, we use $\tilde{W} = diag(\tilde{\mu})$. The *Scoring* algorithm is computationally efficient in its penalized version, since it is in essence the Iteratively Weighted Least Squares algorithm used to estimate a GLM. The algorithm can be generalized to the two-dimensional case in order to estimate $\boldsymbol{\alpha}$.

Let the number of B-splines (m_a and n_y in our case) be $p = n' + q$, with n' the number of equal intervals into which the range of the independent variable is divided ($n'+1$ knots) and $q = 3$.

Thus, when choosing the number of knots, we are actually choosing the number of elements for the B-splines basis. Some proposals to decide about the number of knots for P-splines appear in Eilers and Marx (1996), Currie and Durban (2002) and Ruppert (2002). The latter suggested an empirical rule: for equally spaced data, use one knot for each four or five observations, until reaching a maximum of 40 knots. The resulting number of B-splines is quite smaller than the number of observations. We used that rule in the illustrative applications and carried out the numerical computations with the packages R-2.6.2 and Matlab (version 7.0), see Appendix 3.7.4-3.7.6. Below, we suggest using an iterative algorithm to find the smoothness parameters that yield the desired percentage of smoothness. It should be clear however, that this is not the only way of choosing those values and other procedures might be employed, *e.g.*, by trial and error.

We consider explicitly the one-dimensional case of smoothing by ages. We fix a desired percentage of smoothness by assigning a value to $\mathrm{S}_a(\lambda_a; m)$, recalling that it cannot be higher than 1-2/*m*. Then, since m and K_a are known, we search for the smoothing parameter λ_a that satisfies $1 - tr[(I_m + \lambda_a K'_a K_a)^{-1}]/m = \mathrm{S}_a(\lambda_a; m)$. We start with a low value λ_{a0} and proceed increasing it iteratively by an amount $\Delta \in \mathrm{R}^+$ defined arbitrarily. The process stops when the two sides of the previous equation differ at most by ε (in our case, we used ε = 0.001). In the two-dimensional case, the same index of smoothness can be obtained with different combinations of λ_a and λ_y. If $\mathrm{S}_{ay}(\lambda_a, \lambda_y; m, n)$ is the desired two-dimensional smoothness index (recalling the bounds indicated in Proposition 2), we look for $1 - tr[(I_{mn} + \lambda_a K'_a K_a + \lambda_y K'_y K_y)^{-1}]/mn = \mathrm{S}_{ay}(\lambda_a, \lambda_y; m, n)$ where K_a, K_y, m and n are known quantities. We now have the following options. a) Fix a smoothing parameter in one dimension, say $\lambda_a = \lambda_{a0}$ for ages, and start searching over the values of the other smoothing parameter, as in the one-dimensional case. That is, look for the smoothing parameter that yields the two-dimensional index of smoothness looked for through an iterative process of the form $\lambda_{y\,\text{new}} = \lambda_{y\,\text{last}} + \Delta$, with $\Delta \in \mathrm{R}^+$. The process is stopped when convergence is achieved. b) Run an iterative process in which both parameters, λ_a and λ_y, change in accordance with $\lambda_{a\,\text{new}} = \lambda_{a\,\text{last}} + \Delta_a$ and $\lambda_{y\,\text{new}} = \lambda_{y\,\text{last}} + \Delta_y$, where Δ_a and $\Delta_y \in \mathrm{R}^+$ are defined appropriately. Then, check that the difference between both sides of the equality gets as close to zero as possible (we used as tolerance $\varepsilon = 0.001$).

3.5 Illustrative applications

The following applications use data on mortality for ages 11-100 and years 1947-1999 from the Continuous Mortality Investigation Bureau (CMIB) of the United Kingdom. To illustrate the one-dimensional smoothing situation we consider two possibilities: a single age and different years or different ages and a single year. In the first case, we work with data on mortality at age 65 (in logarithms) for the entire period. The maximum smoothness that can be reached is 96.2% and we present the results for 75% smoothness in Figure 3.2. In the second case, we choose year 1955 and smooth with respect to ages. Figure 2 also presents the corresponding 75% smoothed estimates (the maximum smoothness that can be attained is 97.8%). The variance matrix of the estimates of $\log(d_i/e_i)$ is given by $\mathrm{Var}(\hat{\boldsymbol{y}}_{(\alpha)}) = (\sigma_\eta^{-2} I_m + \sigma_\varepsilon^{-2} N^{-1\prime} D'_2 D_2 N^{-1})^{-1}$. Nevertheless, to take into account the B-spline basis, we used the approximation $\mathrm{V\hat{a}r}(\hat{\boldsymbol{y}}_{(\alpha)}) \approx B(B'WB + \lambda D'_2 D_2)^{-1} B'$ (see Currie *et al.*, 2004, p. 8). Therefore, an interval of ± 3 standard deviations is given by $\log(d_i/e_i) \pm 3\sqrt{\mathrm{V\hat{a}r}(\hat{\boldsymbol{y}}_{(\alpha)i})}$, where $\mathrm{V\hat{a}r}(\hat{\boldsymbol{y}}_{(\alpha)i})$ is the i$^{\text{th}}$ element in the diagonal of $\mathrm{V\hat{a}r}(\hat{\boldsymbol{y}}_{(\alpha)})$.

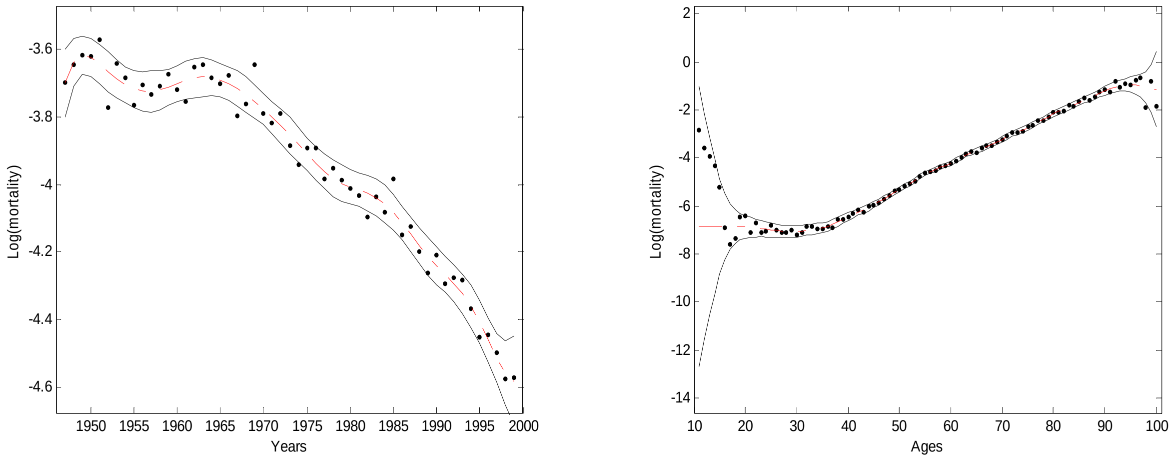


Figure 3.2. Observed and fitted log-mortality with 75% smoothness. Left panel: age 65 (λ =0.1) and right panel: year 1955 (λ = 461), with ± 3 standard deviation intervals.

Since we used the same percentage of smoothness in both graphs of Figure 3.2, we can say that there is more uncertainty in the dimension of years than in ages. For a given age, the decrease in mortality through time evolves too slowly, whereas for a specific year, the whole range of variability in mortality (for all ages) is present. In year 1955, the uncertainty at both ends of the series is greater than in the middle. This is essentially due to the fact that mortality rates at ages 10-25 and greater than 90 have higher variability than at other ages.

For the two-dimensional case we first replicated the smoothing example in Currie *et al.* (2004, pp. 15 and 16) for which AIC = 2306.3, BIC = 4770, λ_a = 0.6, λ_y = 150 and *df* = 41.2. Now we can add that the smoothness attained is 75.6% and the largest percentage of smoothness that can be achieved with this dataset is 97.6%. In Figure 3.3 we appreciate the mortality surfaces corresponding to 75% smoothness obtained with different values of λ_a and λ_y. Thus, we verified empirically that the log-mortality surfaces are different for different combinations of parameters producing the same percentage of smoothness. This is because by setting λ_a = 1 or λ_y = 1 we give priority to smoothing in the other dimension, while in the last case we assign the same parameter values to both dimensions.

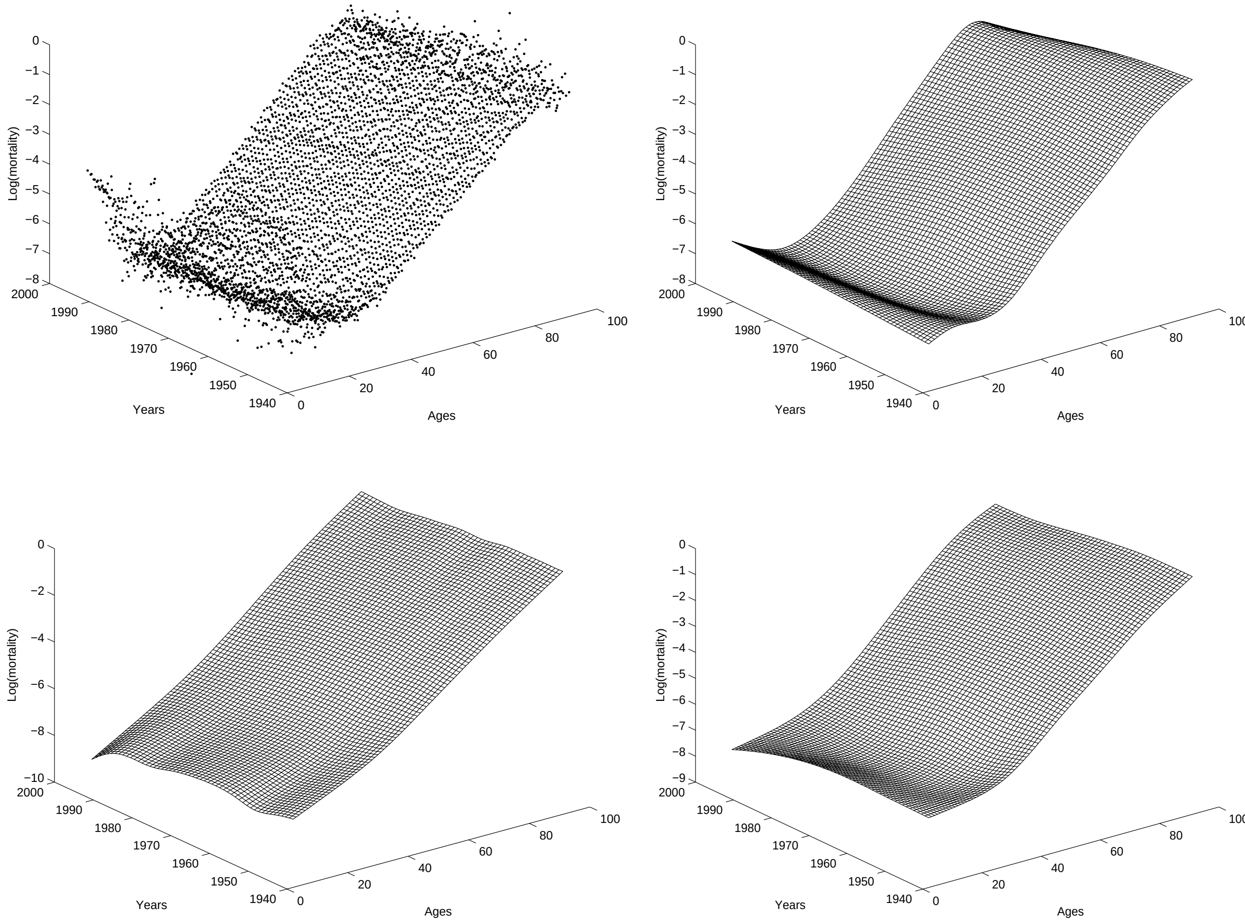


Figure 3.3 Observed log-mortality data (top left) and smoothed by ages 11-100 (top right, $\lambda_a = 1$ and $\lambda_y = 2695$), by years (bottom left, $\lambda_a = 2440$ and $\lambda_y = 1$), and both by ages and years (bottom right, $\lambda_a = \lambda_y = 227$).

In order to link the one-dimensional smoothing solution to the two-dimensional one, we suggest the following. First, fix an index of smoothness in any of the two dimensions; then, on the basis of the maximum percentage of smoothness that can be reached, decide the value of S_{ay}%. Finally, choose the smoothing parameter for the complementary dimension so as to achieve the desired joint smoothness. As an example of this idea, we smoothed the one-dimensional series of ages 30 to 70, for years 1947-1999, with 75% smoothness. In setting the one-dimensional index of smoothness, we used λ_y = 0.1 for all ages (Figure 4 shows the results for age 30). Now, if we look for S_{ay}% = 85% joint smoothness we must find the parameter value λ_a that produces it, given that λ_y = 0.1 is already fixed. An application of the iterative algorithm leads us to choose $\lambda_a = 2165000$ as the value that ensures attainment of the desired two-dimensional percentage of smoothness. The log-mortality surfaces in Figure 4 show the same data as that in Figure 3 from a slightly different perspective that facilitates comparison of the results obtained in two dimensions with respect to those in one dimension. The results for age 30 with the following percentages of smoothness are emphasized: 13.7% (since $\lambda_a \approx 0$), 75% and 85% respectively.

The two charts at the top of Figure 3.4 allow us to compare the smoothing results that may be considered equivalent, although in different dimensions. In fact, the smoothed curves for age 30 show similar dynamics. When we keep the value λ_y = 0.1 and let $\lambda_a \approx 0$, the surface does not change in a sensible manner. This fact was expected since no smoothing is produced in the age dimension. On the other hand, the two graphs at the bottom allow us to see how the smoothness for age 30 becomes more pronounced when the two-dimensional percentage of smoothness increases. We can also have a visual appreciation of the manner in which the log-mortality surface becomes flatter.

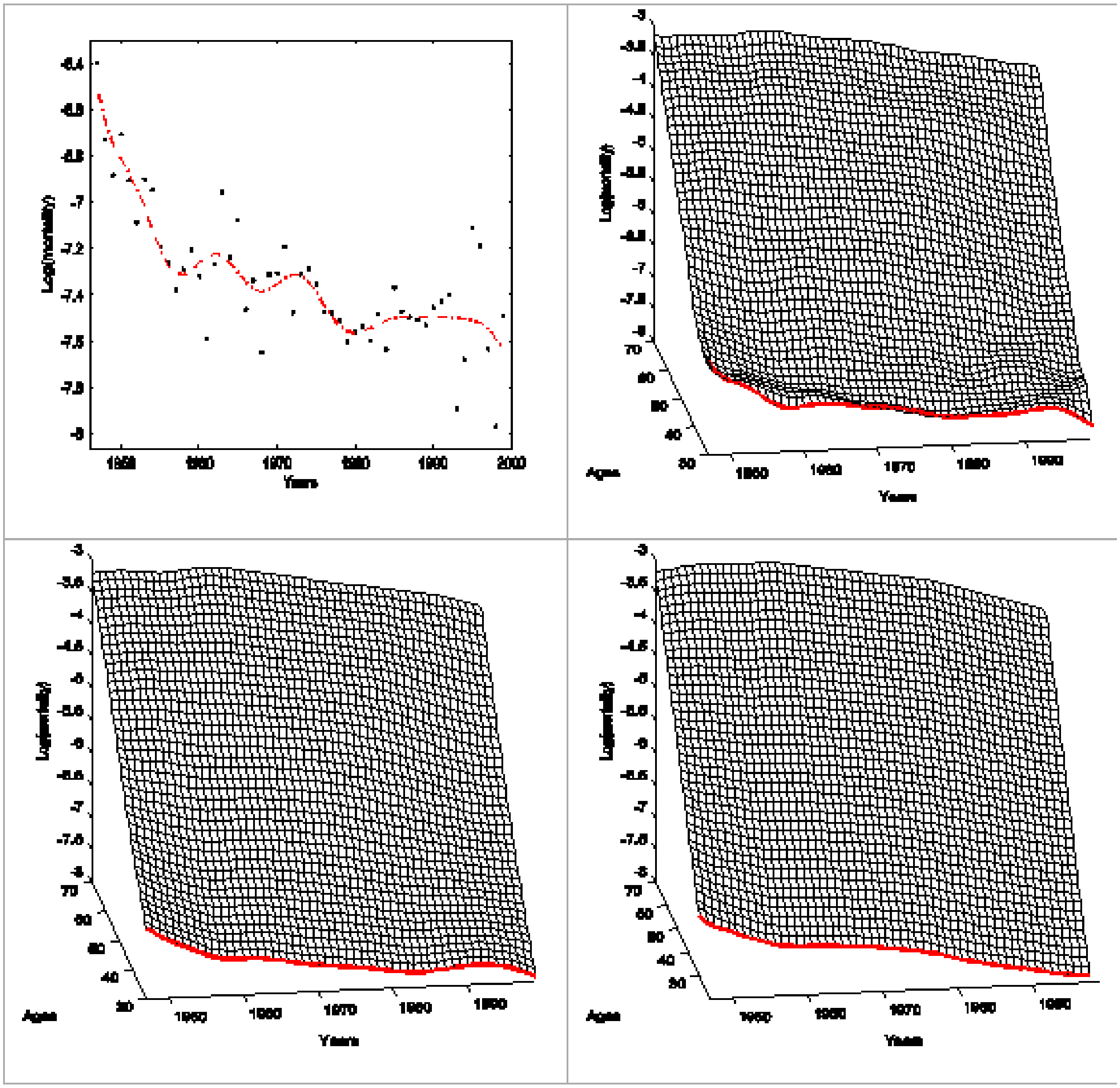


Figure 3.4. Top left: observed and smoothed log-mortality at age 30 and years 1947-1999 with 75% smoothness ($\lambda_y = 0.1$). Smoothed surface by ages 30-70 and different percentages of smoothness: top right 13.7% ($\lambda_a \approx 0$, $\lambda_y = 0.1$); bottom left 75% ($\lambda_a = 5100$, $\lambda_y = 0.1$); bottom right 85% ($\lambda_a = 2165000$, $\lambda_y = 0.1$).

3.6 Conclusions

Our proposal is useful to estimate mortality trends with a desired percentage of smoothness fixed at the outset of the study. By so doing, comparability of trends in mortality rates is enhanced, since it is not strictly valid to compare smoothed estimates with different degrees of smoothness. We can compare smoothed mortality rates by selecting appropriate smoothing parameters with the aid of the proposed index of smoothness, whose properties are established in this work. Our main contribution lies in defining the index of smoothness (that can be calculated from the available data) and showing that it has some desirable properties.

We establish the relationship between one-dimensional and two-dimensional indices of smoothness, and indicate how marginal smoothness can be interpreted. In order to deal with the two-dimensional problem, we basically extend the ideas of one-dimensional smoothing. Our proposal arises from the use of GLS that yields identical results to those already established in the literature. Therefore, we can estimate mortality trends in two dimensions (age and year) using well-known smoothing spline methods. The numerical calculations do not require using high-dimensional arrays, *e.g.* a 5000×5000 B-spline basis and can be performed efficiently by using results that are already established in the field. We recommend fixing a percentage of smoothness first and then applying iterative algorithms to find the smoothing parameters that produce estimates with such degree of smoothness. The empirical examples illustrate the kind of results that can be obtained in practice with our proposal. The case we consider explicitly involves data for which a Poisson distribution is deemed appropriate, but our method can be applied to data with other distributions as well.

It should be stressed that our contribution does not modify the already established theoretical results on P-splines, and that we endorse the use of the computational procedures applied in practice nowadays. What we suggest to change is the way analysts tend to estimate or select the smoothing parameters, when using P-splines.

3.7 Appendix

3.7.1 Proof of Proposition 1.

Straightforward calculations show that the measure of precision share Λ, fulfills the following criteria. Adding-up criterion: if P, Q_a and Q_y are symmetric and positive definite or semidefinite matrices, then $\Lambda(P; P+Q_a+Q_y)+\Lambda(Q_a; P+Q_a+Q_y)+\Lambda(Q_y; P+Q_a+Q_y)=1$. Zero-unit criterion: the limits of the interval (0, 1) are attained if two of the three matrices are the zero matrix, $\Lambda(0; 0+0+Q_y)$ = $\Lambda(0; 0+Q_a+0)$ = 0 and $\Lambda(P; P+0+0)$ = 1. Invariance criterion for nonsingular linear transformations: for a square and nonsingular matrix *L*, $\Lambda(L'PL; L'PL+L'Q_aL+L'Q_yL)$ $=\Lambda(P; P+Q_a+Q_y)$. Linearity criterion: the precision share of $iP_1+jP_2+kP_3$ is i times the share of P_1 plus j times the share of P_2 plus k times the share of P_3, for three triplets of symmetric and positive definite or positive semi-definite matrices P_1, Q_1, R_1, P_2, Q_2, R_2 and P_3, Q_3, R_3 with the same sum $P_1+Q_1+R_1$ = $P_2+Q_2+R_2$ = $P_3+Q_3+R_3$, where $i+j+k$ = 1 with $i, j, k \geq 0$. The proof that Λ is the unique scalar measure fulfilling the four criteria follows directly from the proof provided by Theil (1963) for the case of two matrices *A* and *B,* where the index is given by $\Lambda(A; A+B)$. We only need to recognize that our *P* plays the role of *A* and Q_a + Q_y plays that of *B*.

3.7.2 Proof of Proposition 2.

Let $\gamma_{a,i}$ and $\gamma_{y,j}$ be the eigenvalues of $K'_a K_a$ and $K'_y K_y$, respectively. Then, write $M_{ay} = (I_{mn} + \lambda_a K'_a K_a + \lambda_y K'_y K_y)^{-1}$ for short. Since two eigenvalues of both $K'_a K_a$ and $K'_y K_y$ are zero, we get

$$tr(M_{ay}) = \sum_{j=1}^{m-2}\sum_{i=1}^{n-2}\frac{1}{1+\lambda_a\gamma_{a,i}+\lambda_y\gamma_{y,j}} + 2\sum_{j=1}^{m-2}\frac{1}{1+\lambda_y\gamma_{y,j}} + 2\sum_{i=1}^{n-2}\frac{1}{1+\lambda_a\gamma_{a,i}} + 4.$$

The results follow from the fact that $S_{ay}(\lambda_a, \lambda_y; m, n) = 1 - tr(M_{ay})/mn$, since:

i) If $\lambda_a \to 0$ and $\lambda_y \to 0$, $tr(M_{ay}) \to mn$. ii) If $\lambda_a \to 0$ and λ_y remains constant, $tr(M_{ay}) \to n\left(\sum_{j=1}^{m-2}\frac{1}{1+\lambda_y\gamma_{y,j}} + 2\right)$. iii) If $\lambda_a \to 0$ and $\lambda_y \to \infty$, $tr(M_{ay}) \to 2m$. iv) If λ_a remains constant and $\lambda_y \to 0$, $tr(M_{ay}) \to m\left(\sum_{i=1}^{n-2}\frac{1}{1+\lambda_a\gamma_{a,i}} + 2\right)$. v) If λ_a remains constant and $\lambda_y \to \infty$, $tr(M_{ay}) \to 2\sum_{i=1}^{n-2}\frac{1}{1+\lambda_a\gamma_{a,i}} + 4$. vi) If $\lambda_a \to \infty$ and $\lambda_y \to 0$, $tr(M_{ay}) \to 2n$. vii) If $\lambda_a \to \infty$ and λ_y remains constant, $tr(M_{ay}) \to 2\sum_{j=1}^{m-2}\frac{1}{1+\lambda_y\gamma_{y,j}} + 4$. viii) If $\lambda_a \to \infty$ and $\lambda_y \to \infty$, $tr(M_{ay}) \to 4$.

3.7.3 Proof of Proposition 4.

First, let the smoother matrices be written as $(I_{mn} + \lambda_a K'_a K_a + \lambda_y K'_y K_y)^{-1} = M_{ay}$, $(I_{mn} + \lambda_a K'_a K_a)^{-1} = M_a$ and $(I_{mn} + \lambda_y K'_y K_y)^{-1} = M_y$. We use the following identities (see Kitagawa and Gersch, 1996, p.85) $M_{ay} = M_a - M_a K'_y (K_y M_a K'_y + \lambda_y^{-1} I_{mn})^{-1} K_y M_a$ and $M_a K'_y (K_y M_a K'_y + \lambda_y^{-1} I_{mn})^{-1} = M_{ay} K'_y \lambda_y$. Therefore, we have $M_{ay} = M_a - M_{ay} \lambda_y K'_y K_y M_a$ and by changing the roles played by $\lambda_a K'_a K_a$ and $\lambda_y K'_y K_y$, we get $M_{ay} = M_y - M_{ay} \lambda_a K'_a K_a M_y$.

Hence, when adding the last two equations we get $M_a + M_y = M_{ay}\,(2I_{mn} + \lambda_y\, K'_y\, K_y M_a + \lambda_a\, K'_a\, K_a$

M_y). Now, from Proposition 3 we know that $\boldsymbol{Y}_{(a)} = (I_n \otimes M_a)\boldsymbol{Y}$, hence $tr(I_n \otimes M_a) + tr(I_m \otimes M_y)$

$= tr[M_{ay}(2I_{mn} + \lambda_y\, K'_y\, K_y\, M_a + \lambda_a\, K'_a\, K_a\, M_y)]$ and the result follows from

$$mn[1 - \mathrm{S}_a(\lambda_a; m)] + mn[1 - \mathrm{S}_y(\lambda_y; n)] = 2tr(M_{ay}) + tr[M_{ay}(\lambda_y\, K'_y\, K_y M_a + \lambda_a\, K'_a\, K_a M_y)]$$

$$= 2mn\,[1 - \mathrm{S}_{ay}(\lambda_a,\ \lambda_y; m, n)] + tr[(\lambda_y\, K'_y\, K_y M_a + \lambda_a\, K'_a\, K_a M_y)M_{ay}].$$

3.7.4 R and Matlab routine: One-dimensional case, same age different years

I am indebted to M. Durban for providing some computing programs for this Chapter.

```
#################### Caso univariado ################################

############### R
library(splines)

bspline<-function(x,xl,xr,ndx,bdeg){
dx<-(xr-xl)/ndx
knots<-seq(xl-bdeg*dx,xr+bdeg*dx,by=dx)
B<-spline.des(knots,x,bdeg+1,0*x)$design
B
}
# Ejecutar el generador de datos
#
#     File name: read.s
#     Read data
#
min.age <- 30
max.age <- 30
min.year <- 1947
max.year <- 1999
# stop.year <- trunc((min.year+max.year)/2 + 1)-1#punto medio de rango de datos
stop.year <- 1999
#
#     Claims data
#
D<-read.table("C:\\MATLAB6p5\\work\\Yclaim3Age.dat")
D<-as.matrix(D) #conforma matriz D
D <- t(D)
dimnames(D) <- list(min.age: max.age, min.year: max.year)#etiquetas
d <- c(D)
#
#     Exposure data Yexpo3.dat
#
E <- read.table("C:\\MATLAB6p5\\work\\Yexpo3Age.dat")
E <- as.matrix(E)#conforma matriz E
E <- t(E)
dimnames(E) <- list(min.age: max.age, min.year: max.year)#etiquetas
e <- c(E)
off <- log(e)#se toma logaritmo natural
```

```
#xy<-1:ncol(D)
#ndxy<-10
#By<-bspline(xy,xl=min(xy)-1, xr=max(xy)+1,ndxy,bdeg=3)#número de b-splines = ndx+bdeg

xy<-seq(min.year, max.year,length=max.year-min.year+1)
ndxy<-10
By<-bspline(xy,xl=min(xy)-1,xr=max(xy)+1,ndxy,bdeg=3)#número de b-splines = ndx+bdeg

ny<-ncol(By)
Dy<-diff(diff(diag(ny)))
DtDy<-t(Dy)%*%Dy

V <- matrix(0, nrow = nrow(D), ncol = ncol(D))
dimnames(V) <- list(min.age: max.age, min.year: max.year)
V[ ,1:length(min.year:stop.year)] <- 1
v <- c(V)

eta <- matrix(log( (d+1)/(e+2) ), nrow = nrow(By), ncol = 1)

stopyear <- stop.year
minage <- min.age
maxage <- max.age
minyear <- min.year
maxyear <- max.year

write.table(stopyear,col.names = FALSE, row.names= FALSE, sep = "\t",file="C:\\MATLAB6p5\\work\\stopyear.dat")
write.table(minage,col.names = FALSE, row.names= FALSE, sep = "\t",file="C:\\MATLAB6p5\\work\\minage.dat")
write.table(maxage,col.names = FALSE, row.names= FALSE, sep = "\t",file="C:\\MATLAB6p5\\work\\maxage.dat")
write.table(minyear,col.names = FALSE, row.names= FALSE, sep = "\t",file="C:\\MATLAB6p5\\work\\minyear.dat")
write.table(maxyear,col.names = FALSE, row.names= FALSE, sep = "\t",file="C:\\MATLAB6p5\\work\\maxyear.dat")
write.table(d,col.names = FALSE, row.names= FALSE, sep = "\t",file="C:\\MATLAB6p5\\work\\d.dat")
write.table(e,col.names = FALSE, row.names= FALSE, sep = "\t",file="C:\\MATLAB6p5\\work\\e.dat")
write.table(off,col.names = FALSE, row.names= FALSE, sep = "\t",file="C:\\MATLAB6p5\\work\\off.dat")
write.table(E,col.names = FALSE, row.names= FALSE, sep = "\t",file="C:\\MATLAB6p5\\work\\E.dat")
write.table(xy,col.names = FALSE, row.names= FALSE, sep = "\t",file="C:\\MATLAB6p5\\work\\xy.dat")
write.table(DtDy,col.names = FALSE, row.names= FALSE, sep = "\t",file="C:\\MATLAB6p5\\work\\DtDy.dat")
write.table(By,col.names = FALSE, row.names= FALSE, sep = "\t",file="C:\\MATLAB6p5\\work\\By.dat")
write.table(ny,col.names = FALSE, row.names= FALSE, sep = "\t",file="C:\\MATLAB6p5\\work\\ny.dat")
write.table(eta,col.names = FALSE, row.names= FALSE, sep = "\t",file="C:\\MATLAB6p5\\work\\eta.dat")
write(D, file = "C:\\MATLAB6p5\\work\\D.dat",ncolumns = 1, sep = "\t")
write(v, file="C:\\MATLAB6p5\\work\\v.dat", ncolumns = 1, sep = "\t")

%%%%%%%%MATLAB
clear all
format long

load C:\MATLAB6p5\work\v.dat
load C:\MATLAB6p5\work\D.dat
load C:\MATLAB6p5\work\stopyear.dat
load C:\MATLAB6p5\work\minage.dat
load C:\MATLAB6p5\work\maxage.dat
load C:\MATLAB6p5\work\minyear.dat
load C:\MATLAB6p5\work\maxyear.dat
load C:\MATLAB6p5\work\d.dat
load C:\MATLAB6p5\work\e.dat
load C:\MATLAB6p5\work\off.dat
load C:\MATLAB6p5\work\E.dat
load C:\MATLAB6p5\work\xy.dat
load C:\MATLAB6p5\work\DtDy.dat
load C:\MATLAB6p5\work\By.dat
load C:\MATLAB6p5\work\ny.dat
```

```
load C:\MATLAB6p5\work\eta.dat

ainit=inv(By'*By)*By'*eta;
y=D;
n=max(size(y));

lambday = 10;
lambda2 = abs(lambday);
P=DtDy;

a = ainit;
aold = 10;
iter = 0;
tol = 1;
TOL=10^(-8);
MAXITER=20;

tic
while(tol > TOL & iter < MAXITER)
  iter = iter + 1;
  Ita = off + By * a;                  % 4770x1
  Mu = exp(Ita);                       % 4770x1
  Wt = Mu.*v;                          % 4770x1
  BtWB = By' * (diag(Wt)* By);         % 169x169
  Rhs = BtWB * a + By' * (v.*(y - Mu)); % 169x1 + 169x4770*(4770x1)
  anew= inv(BtWB + lambda2*P)*Rhs;               % 169x1
  tol = max(abs(anew - aold))/mean(abs(anew));
  aold=a;
  a=anew;
end
toc

  Ita = off + By * a;
  Mu = exp(Ita);
  Wt = Mu.*v;
  BtWB = By'*(diag(Wt)*By); %
  BtWBplusP = BtWB + lambda2*P;
  Rhs = BtWB * a + By' * (v.*(y - Mu));
  a= inv(BtWBplusP)*Rhs;
  Tr = trace(inv(BtWBplusP)*BtWB);
  yinit = y;
  for i=1:max(size(y))
    if yinit(i) == 0
        yinit(i) = 10^(-4);
    end
  end
  Dev = 2*sum(sum(v.*y.*log(yinit./Mu)));
  Bic = Dev + log(sum(sum((v)))) * Tr;
  Aic = (Dev/2) + (2*Tr);
  Hazard = Ita - off;

sua = 1-(Tr/n) %suavidad generada con el lambda dado inicialmente
display('    lambday              Tr                suavidad')
salida1=[lambda2 Tr sua]

load('C:\MATLAB6p5\work\Ylogei3Age.dat')
logei=Ylogei3Age;
logMu=logei+By*a;
%Wt=diag(logMu);
%VarBy=(By*inv(BtWBplusP)*By'*Wt);%suponiendo varianza 1 en el ECM
%dVar=diag(VarBy);
```

```
figure
plot(xy,eta,'k.')
xlabel('Years');
ylabel('Log(mortality)');
xlim([min(xy)-1 max(xy)+1])
ylim([min(eta)-0.1 max(eta)+0.1])

clear lb   %limpia la variable lambda
clear suaf %limpia la variable de suavidad final deseada por el usuario

suaf= 0.75  %suavidad final deseada por el usuario
nmin=2/(1-suaf)  %número mínimo de observaciones para aspirar a la suavidad deseada
cs=1-(2/n)       %máxima suavidad que se puede alcanzar

tic
lb = 0.1;
parc=  1-(trace(By*inv(By'*diag(Wt)*By+0*P)*By'*diag(Wt))/n);
while (1-(trace(By*inv(By'*diag(Wt)*By+lb*P)*By'*diag(Wt)))/n)<=suaf
        lb = lb + 0.1;
        parc=[parc (1-(trace(By*inv(By'*diag(Wt)*By+lb*P)*By'*diag(Wt)))/n)];
        %parc=[parc (1-(trace(By*inv(By'*Wt*By+lb*P)*By'*Wt))/n)];
end
toc
lb=lb
parc=parc';
parc=parc(1:max(size(parc')-1));

%Gráfica con los datos de acuerdo con la suavidad solicitada por el usuario
%aest=inv(By'*Wt*By+lb*P)*By'*Wt*eta;
aest=inv(By'*diag(Wt)*By+lb*P)*By'*diag(Wt)*eta;
hold on
plot(xy,By*aest,'r-.')
xlim([min(xy)-1 max(xy)+1])
ylim([min(eta)-0.1 max(eta)+0.1])
display('    lambday              suavidad')
salida2=[lb suaf]

Var=(By*inv(BtWBplusP)*By');%
dVar=sqrt(diag(Var));

% límite superior
ls=(By*aest)+3*dVar;
hold on
plot(xy,ls,'k-')

% límite inferior
li=(By*aest)-3*dVar;
hold on
plot(xy,li,'k-')
```

3.7.5 R and Matlab routine: One-dimensional case, same year different ages

```
#################### Caso univariado #################################
############### R
library(splines)

bspline<-function(x,xl,xr,ndx,bdeg){
dx<-(xr-xl)/ndx
knots<-seq(xl-bdeg*dx,xr+bdeg*dx,by=dx)
B<-spline.des(knots,x,bdeg+1,0*x)$design
```

```
B
}
#
#     File name: read.s
#     Read data
#
min.age <- 11
max.age <- 100
min.year <- 1947
max.year <- 1947
# stop.year <- trunc((min.year+max.year)/2 + 1)-1#punto medio de rango de datos
stop.age <- 100
#stop.year <-1974

# Ejecutar el generador de datos
#
#     Claims data
#
D<-read.table("C:\\MATLAB6p5\\work\\Yclaim3Year.dat")
D<-as.matrix(D) #conforma matriz D
D <- t(D)
dimnames(D) <- list(min.year: max.year, min.age: max.age)#etiquetas
d <- c(D)
#
#     Exposure data Yexpo3.dat
#
E <- read.table("C:\\MATLAB6p5\\work\\Yexpo3Year.dat")
E <- as.matrix(E)#conforma matriz E
E <- t(E)
dimnames(E) <- list(min.year: max.year, min.age: max.age)#etiquetas
e <- c(E)
off <- log(e)#se toma logaritmo natural

xa<-seq(min.age, max.age,length=max.age-min.age+1)
ndxa<-10 #length(xa)-3
Ba<-bspline(xa,xl=min(xa),xr=max(xa),ndxa,bdeg=3)#número de b-splines = ndx+bdeg

na<-ncol(Ba)
Da<-diff(diff(diag(na)))
DtDa<-t(Da)%*%Da

V <- matrix(0, nrow = nrow(D), ncol = ncol(D))
dimnames(V) <- list(min.year: max.year, min.age: max.age)
V[ ,1:length(min.age:stop.age)] <- 1
#V[ ,1:length(min.year:stop.year)] <- 1
v <- c(V)

eta <- matrix(log( (d+1)/(e+2) ), nrow = nrow(Ba), ncol = 1)

stopage <- stop.age
#stopyear <- stop.year
minage <- min.age
maxage <- max.age
minyear <- min.year
maxyear <- max.year

write.table(stopage,col.names = FALSE, row.names= FALSE, sep = "\t",file="C:\\MATLAB6p5\\work\\stopage.dat")
#write.table(stopyear,col.names = FALSE, row.names= FALSE, sep = "\t",file="C:\\MATLAB6p5\\work\\stopyear.dat")
write.table(minage,col.names = FALSE, row.names= FALSE, sep = "\t",file="C:\\MATLAB6p5\\work\\minage.dat")
write.table(maxage,col.names = FALSE, row.names= FALSE, sep = "\t",file="C:\\MATLAB6p5\\work\\maxage.dat")
write.table(minyear,col.names = FALSE, row.names= FALSE, sep = "\t",file="C:\\MATLAB6p5\\work\\minyear.dat")
```

```
write.table(maxyear,col.names = FALSE, row.names= FALSE, sep = "\t",file="C:\\MATLAB6p5\\work\\maxyear.dat")
write.table(d,col.names = FALSE, row.names= FALSE, sep = "\t",file="C:\\MATLAB6p5\\work\\d.dat")
write.table(e,col.names = FALSE, row.names= FALSE, sep = "\t",file="C:\\MATLAB6p5\\work\\e.dat")
write.table(off,col.names = FALSE, row.names= FALSE, sep = "\t",file="C:\\MATLAB6p5\\work\\off.dat")
write.table(E,col.names = FALSE, row.names= FALSE, sep = "\t",file="C:\\MATLAB6p5\\work\\E.dat")
write.table(xa,col.names = FALSE, row.names= FALSE, sep = "\t",file="C:\\MATLAB6p5\\work\\xa.dat")
write.table(DtDa,col.names = FALSE, row.names= FALSE, sep = "\t",file="C:\\MATLAB6p5\\work\\DtDa.dat")
write.table(Ba,col.names = FALSE, row.names= FALSE, sep = "\t",file="C:\\MATLAB6p5\\work\\Ba.dat")
write.table(na,col.names = FALSE, row.names= FALSE, sep = "\t",file="C:\\MATLAB6p5\\work\\na.dat")
write.table(eta,col.names = FALSE, row.names= FALSE, sep = "\t",file="C:\\MATLAB6p5\\work\\eta.dat")
write(D, file = "C:\\MATLAB6p5\\work\\D.dat",ncolumns = 1, sep = "\t")
write(v, file="C:\\MATLAB6p5\\work\\v.dat", ncolumns = 1, sep = "\t")

%%%%%%%%MATLAB
clear all
format long

load C:\MATLAB6p5\work\v.dat
load C:\MATLAB6p5\work\D.dat
%load C:\MATLAB6p5\work\stopyear.dat
load C:\MATLAB6p5\work\stopage.dat
load C:\MATLAB6p5\work\minage.dat
load C:\MATLAB6p5\work\maxage.dat
load C:\MATLAB6p5\work\minyear.dat
load C:\MATLAB6p5\work\maxyear.dat
load C:\MATLAB6p5\work\d.dat
load C:\MATLAB6p5\work\e.dat
load C:\MATLAB6p5\work\off.dat
load C:\MATLAB6p5\work\E.dat
load C:\MATLAB6p5\work\xa.dat
load C:\MATLAB6p5\work\DtDa.dat
load C:\MATLAB6p5\work\Ba.dat
load C:\MATLAB6p5\work\na.dat
load C:\MATLAB6p5\work\eta.dat

ainit=inv(Ba'*Ba)*Ba'*eta;
y=D;
n2=max(size(y));

lambdaa = 29;
lambda1 = abs(lambdaa);
P=DtDa;

a = ainit;
aold = 10;
iter = 0;
tol = 1;
TOL=10^(-8);
MAXITER=20;

tic
while(tol > TOL & iter < MAXITER)
  iter = iter + 1;
  Ita = off + Ba * a;                  % 4770x1
  Mu = exp(Ita);                       % 4770x1
  Wt = Mu.*v;                          % 4770x1
  BtWB = Ba' * (diag(Wt)* Ba);           % 169x169
  Rhs = BtWB * a + Ba' * (v.*(y - Mu)); % 169x1 + 169x4770*(4770x1)
  anew= inv(BtWB + lambda1*P)*Rhs;               % 169x1
  tol = max(abs(anew - aold))/mean(abs(anew));
  aold=a;
```

```
  a=anew;
end
toc

  Ita = off + Ba * a; %4770           1
  Mu = exp(Ita); % 4770           1
  Wt = Mu.*v; % son vectores cada uno y genera un vector columna
  BtWB = Ba'*(diag(Wt)*Ba); %
  BtWBplusP = BtWB + lambda1*P;
  Rhs = BtWB * a + Ba' * (v.*(y - Mu));
  a= inv(BtWBplusP)*Rhs;
  Tr = trace(inv(BtWBplusP)*BtWB);
  yinit = y;
  for i=1:max(size(y))
    if yinit(i) == 0
        yinit(i) = 10^(-4);
    end
  end
  Dev = 2*sum(sum(v.*y.*log(yinit./Mu)));
  Bic = Dev + log(sum(sum((v)))) * Tr;
  Aic = (Dev/2) + (2*Tr);
  Hazard = Ita - off;

sua = 1-(Tr/n2); %suavidad generada con el lambda dado inicialmente
display('    lambdaa               Tr                 suavidad')
salida1=[lambda1 Tr sua]

load('C:\MATLAB6p5\work\Ylogei3Year.dat')
logei=Ylogei3Year;
logMu=logei+Ba*a;

figure
plot(xa,eta,'k.')
xlabel('Ages');
ylabel('Log(mortality)');
xlim([min(xa)-1 max(xa)+1])
ylim([min(eta)-0.1 max(eta)+0.1])

clear lb   %limpia la variable lambda
clear suaf %limpia la variable de suavidad final deseada por el usuario

suaf= 0.75;  %suavidad final deseada por el usuario
nmin=2/(1-suaf)  %número mínimo de observaciones para aspirar a la suavidad deseada
cs=1-(2/n2)       %máxima suavidad que se puede alcanzar

tic
lb = 1;
parc=  1-(trace(Ba*inv(Ba'*diag(Wt)*Ba+0*P)*Ba'*diag(Wt))/n2);
while (1-(trace(Ba*inv(Ba'*diag(Wt)*Ba+lb*P)*Ba'*diag(Wt)))/n2)<=suaf
            lb = lb + 1;
            parc=[parc (1-(trace(Ba*inv(Ba'*diag(Wt)*Ba+lb*P)*Ba'*diag(Wt)))/n2)];
end
toc
lb
parc=parc';
parc=parc(1:max(size(parc')-1));

%Gráfica con los datos de acuerdo con la suavidad solicitada por el usuario
aest=inv(Ba'*diag(Wt)*Ba+lb*P)*Ba'*diag(Wt)*eta;
hold on
plot(xa,Ba*aest,'r-.')
```

```
xlim([min(xa)-1 max(xa)+1])
ylim([min(eta)-7 max(eta)+3])
salida2=[lb suaf]

Var=(Ba*inv(BtWBplusP)*Ba');%
dVar=sqrt(diag(Var));

% límite superior
ls=(Ba*aest)+3*dVar;
hold on
plot(xa,ls,'k-')

% límite inferior
li=(Ba*aest)-3*dVar;
hold on
plot(xa,li,'k-')
```

3.7.6 R and Matlab routine: Two-dimensional case

```
##################### Caso bivariado ###############################
###################### R
[z]=gendatcon(21,30,1947,1999)
library(splines)
bspline<-function(x,xl,xr,ndx,bdeg){
dx<-(xr-xl)/ndx
knots<-seq(xl-bdeg*dx,xr+bdeg*dx,by=dx)
B<-spline.des(knots,x,bdeg+1,0*x)$design
B
}

#
#	Parameters
#
min.age <- 21
max.age <- 30
min.year <- 1947
max.year <- 1999
stop.year <- 1974
#
#	Claims data
#
D <- scan("C:\\MATLAB6p5\\work\\Yclaim3.dat")# Segmentos de datos
D <- matrix(D, nrow = (max.age - min.age + 1),
		ncol = (max.year - min.year + 1))
dimnames(D) <- list(min.age: max.age, min.year: max.year)#etiquetas

#D <- scan("C:\\2_ST\\R\\Claims.dat", sep = " ")# Total de datos MD
#D <- matrix(D, nrow = (max.year - min.year + 1),
#		ncol = (max.age - min.age + 1))#conforma matriz D
#D <- t(D)
#dimnames(D) <- list(min.age: max.age, min.year: max.year)#etiquetas
d <- c(D)
#
#	Exposure data
#
E <- scan("C:\\MATLAB6p5\\work\\Yexpo3.dat")# Segmentos de datos
E <- matrix(E, nrow = (max.age - min.age + 1),
		ncol = (max.year - min.year + 1))
```

```
dimnames(E) <- list(min.age: max.age, min.year: max.year)#etiquetas

#E <- scan("C:\\2_ST\\R\\Exposure.dat", sep = " ")# Total de datos MD
#E <- matrix(E, nrow = (max.year - min.year + 1),
#            ncol = (max.age - min.age + 1))#conforma matriz E
#E <- t(E)
#dimnames(E) <- list(min.age: max.age, min.year: max.year)#etiquetas
e <- c(E)
off <- log(e)

xa<-1:nrow(D)
ndxa<-10
#ndxa<-length(xa)-3
Ba<-bspline(xa,xl=min(xa)-1, xr=max(xa)+1,ndxa,bdeg=3)#número de b-splines = ndx+bdeg

na<-ncol(Ba)
Da<-diff(diff(diag(na)))
DtDa<-t(Da)%*%Da

xy<-1:ncol(D)
ndxy<-10
#ndxy<-length(xy)-3
By<-bspline(xy,xl=min(xy)-1, xr=max(xy)+1,ndxy,bdeg=3)#número de b-splines = ndx+bdeg

ny<-ncol(By)
Dy<-diff(diff(diag(ny)))
DtDy<-t(Dy)%*%Dy

B <- kronecker(By, Ba)
Pa <- kronecker(diag(ncol(By)), DtDa)
Py <- kronecker(DtDy, diag(ncol(Ba)))

V <- matrix(0, nrow = nrow(D), ncol = ncol(D))
dimnames(V) <- list(min.age: max.age, min.year: max.year)
V[ ,1:length(min.year:stop.year)] <- 1
v <- c(V)

eta <- matrix(log((d+1)/(e+2)), nrow = nrow(B), ncol = 1)

stopyear <- stop.year
minage <- min.age
maxage <- max.age
minyear <- min.year
maxyear <- max.year

write.table(na,col.names = FALSE, row.names= FALSE, sep = "\t",file="D:\\MATLAB6p5\\work\\na.dat")
write.table(ny,col.names = FALSE, row.names= FALSE, sep = "\t",file="D:\\MATLAB6p5\\work\\ny.dat")
write.table(v,col.names = FALSE, row.names= FALSE, sep = "\t",file="D:\\MATLAB6p5\\work\\v.dat")
write.table(stopyear,col.names = FALSE, row.names= FALSE, sep = "\t",file="D:\\MATLAB6p5\\work\\stopyear.dat")
write.table(minage,col.names = FALSE, row.names= FALSE, sep = "\t",file="D:\\MATLAB6p5\\work\\minage.dat")
write.table(maxage,col.names = FALSE, row.names= FALSE, sep = "\t",file="D:\\MATLAB6p5\\work\\maxage.dat")
write.table(minyear,col.names = FALSE, row.names= FALSE, sep = "\t",file="D:\\MATLAB6p5\\work\\minyear.dat")
write.table(maxyear,col.names = FALSE, row.names= FALSE, sep = "\t",file="D:\\MATLAB6p5\\work\\maxyear.dat")
write.table(d,col.names = FALSE, row.names= FALSE, sep = "\t",file="D:\\MATLAB6p5\\work\\d.dat")
write.table(e,col.names = FALSE, row.names= FALSE, sep = "\t",file="D:\\MATLAB6p5\\work\\e.dat")
write.table(off,col.names = FALSE, row.names= FALSE, sep = "\t",file="D:\\MATLAB6p5\\work\\off.dat")
write.table(E,col.names = FALSE, row.names= FALSE, sep = "\t",file="D:\\MATLAB6p5\\work\\E.dat")
write.table(xa,col.names = FALSE, row.names= FALSE, sep = "\t",file="D:\\MATLAB6p5\\work\\xa.dat")
write.table(Ba,col.names = FALSE, row.names= FALSE, sep = "\t",file="D:\\MATLAB6p5\\work\\Ba.dat")
write.table(DtDa,col.names = FALSE, row.names= FALSE, sep = "\t",file="D:\\MATLAB6p5\\work\\DtDa.dat")
write.table(Pa,col.names = FALSE, row.names= FALSE, sep = "\t",file="D:\\MATLAB6p5\\work\\Pa.dat")
```

```
write.table(xy,col.names = FALSE, row.names= FALSE, sep = "\t",file="D:\\MATLAB6p5\\work\\xy.dat")
write.table(DtDy,col.names = FALSE, row.names= FALSE, sep = "\t",file="D:\\MATLAB6p5\\work\\DtDy.dat")
write.table(By,col.names = FALSE, row.names= FALSE, sep = "\t",file="D:\\MATLAB6p5\\work\\By.dat")
write.table(Py,col.names = FALSE, row.names= FALSE, sep = "\t",file="D:\\MATLAB6p5\\work\\Py.dat")
write.table(ny,col.names = FALSE, row.names= FALSE, sep = "\t",file="D:\\MATLAB6p5\\work\\ny.dat")
write.table(na,col.names = FALSE, row.names= FALSE, sep = "\t",file="D:\\MATLAB6p5\\work\\na.dat")
write.table(eta,col.names = FALSE, row.names= FALSE, sep = "\t",file="D:\\MATLAB6p5\\work\\eta.dat")
write.table(B,col.names = FALSE, row.names= FALSE, sep = "\t",file="D:\\MATLAB6p5\\work\\B.dat")
write(D, file = "D:\\MATLAB6p5\\work\\D.dat",ncolumns = 1, sep = "\t")
write(v, file="D:\\MATLAB6p5\\work\\v.dat", ncolumns = 1, sep = "\t")

%%%%%%%MATLAB
clear all
format long

load D:\MATLAB6p5\work\na.dat
load D:\MATLAB6p5\work\ny.dat
load D:\MATLAB6p5\work\v.dat
load D:\MATLAB6p5\work\D.dat
load D:\MATLAB6p5\work\stopyear.dat
load D:\MATLAB6p5\work\minage.dat
load D:\MATLAB6p5\work\maxage.dat
load D:\MATLAB6p5\work\minyear.dat
load D:\MATLAB6p5\work\maxyear.dat
load D:\MATLAB6p5\work\d.dat
load D:\MATLAB6p5\work\e.dat
load D:\MATLAB6p5\work\off.dat
load D:\MATLAB6p5\work\E.dat
load D:\MATLAB6p5\work\xa.dat
load D:\MATLAB6p5\work\Ba.dat
load D:\MATLAB6p5\work\DtDa.dat
load D:\MATLAB6p5\work\Pa.dat
load D:\MATLAB6p5\work\xy.dat
load D:\MATLAB6p5\work\DtDy.dat
load D:\MATLAB6p5\work\By.dat
load D:\MATLAB6p5\work\Py.dat
load D:\MATLAB6p5\work\ny.dat
load D:\MATLAB6p5\work\na.dat
load D:\MATLAB6p5\work\eta.dat
load D:\MATLAB6p5\work\B.dat

tBB=(sparse(B')*sparse(B));
tBeta=sparse(B')*eta;
ainit=tBB\tBeta;
y=D;

lambdaa = 0.6;
lambday = 150;
lambda1 = abs(lambdaa);
lambda2 = abs(lambday);
P = lambda1 * Pa + lambda2 * Py; %169   169

a = ainit;
aold = 10;
iter = 0;
tol = 1;
TOL=10^(-8);
MAXITER=20;

while(tol > TOL & iter < MAXITER)
  iter = iter + 1;
```

```
  Ita = off + sparse(B) * a;                    % 4770x1
  Mu = exp(Ita);                        % 4770x1
  Wt = Mu.*v;                           % 4770x1
  BtWB = sparse(B') * (diag(Wt)* sparse(B));           % 169x169
  Rhs = BtWB * a + sparse(B') * (v.*(y - Mu)); % 169x1 + 169x4770*(4770x1)
  %anewprueba=sparse(BtWB + P);
  %anew= inv(BtWB + P)*Rhs;               % 169x1
  anewp= (BtWB + P);               % 169x1
  anew= anewp\Rhs;               % 169x1
  tol = max(abs(anew - aold))/mean(abs(anew));
  aold=a;
  a=anew;
end

  Ita = off + sparse(B) * a;
  Mu = exp(Ita);
  Wt = Mu.*v; % son vectores cada uno y genera un vector columna
  BtWB = sparse(B')*(diag(Wt)*sparse(B)); %
  BtWBplusP = BtWB + P;
  Rhs = BtWB * a + sparse(B') * (v.*(y - Mu));
  %a= inv(BtWBplusP)*Rhs;
  a= BtWBplusP\Rhs;
  %Tr = trace(inv(BtWBplusP)*BtWB);
  Tr = trace(BtWBplusP\BtWB);
  yinit = y;
  for i=1:max(size(y))
    if yinit(i) == 0
       yinit(i) = 10^(-4);
    end
  end
  Dev = 2*sum(sum(v.*y.*log(yinit./Mu)));
  Bic = Dev + log(sum(sum((v)))) * Tr;
  Aic = (Dev/2) + (2*Tr);
  Hazard = Ita - off;

%%Gráficas
y1=minyear;
yn=maxyear;
a1=minage;
an=maxage;
da=an-a1+1;
dy=yn-y1+1;

load D:\MATLAB6p5\work\edadanos.txt
edadanos;
[ran]=find(edadanos(:,1)>=a1 & edadanos(:,1)<=an & edadanos(:,2)>=y1 & edadanos(:,2)<=yn);
rango=edadanos(ran,:);
age=rango(:,1);
year=rango(:,2);

datos=[edadanos(ran,:) Hazard];
datos2=sortrows(datos);
age2=datos2(:,1);
year2=datos2(:,2);
Hazard2=datos2(:,3);

%Con puntos solamente
%sin suavizar
figure
plot3(age,year,eta,'k.')
%suavizados
```

```
figure
plot3(age,year,Hazard,'k.')

%Con lineas solamente
figure
plot3(age(1:da),year(1:da),Hazard(1:da),'k-')
hold on
plot3(age2(1:dy),year2(1:dy),Hazard2(1:dy),'k-')
hold on
for i=2:dy
        plot3(age(1+da*(i-1):da*i),year(1+da*(i-1):da*i),Hazard(1+da*(i-1):da*i),'k-')
        hold on
end
for j=2:da
    plot3(age2(1+dy*(j-1):dy*j),year2(1+dy*(j-1):dy*j),Hazard2(1+dy*(j-1):dy*j),'k-')
        hold on
end

%Con lineas y puntos observados
figure
plot3(age(1:da),year(1:da),Hazard(1:da),'k-')
hold on
plot3(age2(1:dy),year2(1:dy),Hazard2(1:dy),'k-')
hold on
for i=2:dy
        plot3(age(1+da*(i-1):da*i),year(1+da*(i-1):da*i),Hazard(1+da*(i-1):da*i),'k-')
        hold on
        plot3(age(1+da*(i-1):da*i),year(1+da*(i-1):da*i),eta(1+da*(i-1):da*i),'k.')
        hold on
end
for j=2:da
    plot3(age2(1+dy*(j-1):dy*j),year2(1+dy*(j-1):dy*j),Hazard2(1+dy*(j-1):dy*j),'k-')
        hold on
end

suac=1-(Tr/(na*ny)); %suavidad inicialmente generada
suacmax=1-(4/(na*ny)); %suavidad máxima por aspirar
display('    Aic     Bic     lambdaa     lambday     Tr     suavidad');
salida=[Aic Bic lambda1 lambda2 Tr suac]
toc

%%%%%%%%%%%%%%%%%%%%%%%%
%Suavidad conjunta

clear lba  %limpia la variable lba
clear lby  %limpia la variable lby
clear P2   %limpia la variable P2
clear suaf %limpia la variable de suavidad final deseada por el usuario

%opc=1; suavidad conjunta dada lba
%opc=2; suavidad conjunta dada lby
%opc=3; suavidad conjunta con lby a lba partiendo de cero

opc=1;
tic
lba = 0.56;
lby = 1;
P2 = lba * Pa + lby * Py;
BtWBplusP2 = BtWB + P2;
Tr2 = trace(inv(BtWBplusP2)*BtWB);
```

```
suaf= 0.90;  % 0.75643922465; suavidad conjunta elegida por el usuario
parc=  (1-(Tr2/(na*ny)));

while (1-(Tr2/(na*ny)))<=suaf
        if opc==1
                lby = lby + 1;
                lba = lba;
    elseif opc==2
                lba = lba + 1;
                lby = lby;
        elseif opc==3
                lby=lby+1;
                lby=lba+1;
        end
        P2 = lba * Pa + lby * Py;
        BtWBplusP2 = BtWB + P2;
        Tr2 = trace(inv(BtWBplusP2)*BtWB);
        parc=[parc (1-(Tr2/(na*ny)))];
end
toc
lby=lby
lba=lba
parc=parc';
parc=parc(1:max(size(parc')-1))

%Nuevos datos suavizados
a2 = ainit;
aold2 = 10;
iter2 = 0;
tol2 = 1;
TOL2=10^(-8);
MAXITER2=20;

tic
while(tol2 > TOL2 & iter2 < MAXITER2)
  iter2 = iter2 + 1;
  Ita2 = off + B * a2;                  % 4770x1
  Mu2 = exp(Ita2);                      % 4770x1
  Wt2 = Mu2.*v;                         % 4770x1
  BtWB = B' * (diag(Wt2)* B);           % 169x169
  Rhs2 = BtWB * a2 + B' * (v.*(y - Mu2)); % 169x1 + 169x4770*(4770x1)
  anew2= inv(BtWB + P2)*Rhs2;              % 169x1
  tol2 = max(abs(anew2 - aold2))/mean(abs(anew2));
  aold2=a2;
  a2=anew2;
end
toc

Ita2 = off + B * a2;
Haz = Ita2 - off;

datos22=[edadanos(ran,:) Haz];
datos222=sortrows(datos22);
Haz2=datos222(:,3);

%Con lineas ambas
figure
plot3(age(1:da),year(1:da),Hazard(1:da),'k-')
hold on
plot3(age2(1:dy),year2(1:dy),Hazard2(1:dy),'k-')
hold on
```

```
plot3(age(1:da),year(1:da),Haz(1:da),'b-')
hold on
plot3(age2(1:dy),year2(1:dy),Haz2(1:dy),'b-')
hold on
for i=2:dy
		plot3(age(1+da*(i-1):da*i),year(1+da*(i-1):da*i),Hazard(1+da*(i-1):da*i),'k-')
		hold on
end
for j=2:da
	plot3(age2(1+dy*(j-1):dy*j),year2(1+dy*(j-1):dy*j),Hazard2(1+dy*(j-1):dy*j),'k-')
		hold on
end
for i=2:dy
		plot3(age(1+da*(i-1):da*i),year(1+da*(i-1):da*i),Haz(1+da*(i-1):da*i),'b-')
		hold on
end
for j=2:da
	plot3(age2(1+dy*(j-1):dy*j),year2(1+dy*(j-1):dy*j),Haz2(1+dy*(j-1):dy*j),'b-')
		hold on
end

%Con lineas solamente (suavizada con elección del usuario)
figure
plot3(age(1:da),year(1:da),Haz(1:da),'b-')
hold on
plot3(age2(1:dy),year2(1:dy),Haz2(1:dy),'b-')
hold on
for i=2:dy
		plot3(age(1+da*(i-1):da*i),year(1+da*(i-1):da*i),Haz(1+da*(i-1):da*i),'b-')
		hold on
end
for j=2:da
	plot3(age2(1+dy*(j-1):dy*j),year2(1+dy*(j-1):dy*j),Haz2(1+dy*(j-1):dy*j),'b-')
		hold on
end

%Con lineas solamente (suavizada sin elección del usuario)
figure
plot3(age(1:da),year(1:da),Hazard(1:da),'k-')
hold on
plot3(age2(1:dy),year2(1:dy),Hazard2(1:dy),'k-')
hold on
for i=2:dy
		plot3(age(1+da*(i-1):da*i),year(1+da*(i-1):da*i),Hazard(1+da*(i-1):da*i),'k-')
		hold on
end
for j=2:da
	plot3(age2(1+dy*(j-1):dy*j),year2(1+dy*(j-1):dy*j),Hazard2(1+dy*(j-1):dy*j),'k-')
		hold on
end

%Con lineas y puntos observados (suavizada con elección del usuario)
figure
plot3(age(1:da),year(1:da),Haz(1:da),'b-')
hold on
plot3(age2(1:dy),year2(1:dy),Haz2(1:dy),'b-')
hold on
for i=2:dy
		plot3(age(1+da*(i-1):da*i),year(1+da*(i-1):da*i),Haz(1+da*(i-1):da*i),'b-')
		hold on
		plot3(age(1+da*(i-1):da*i),year(1+da*(i-1):da*i),eta(1+da*(i-1):da*i),'k.')
```

```
        hold on
end
for j=2:da
    plot3(age2(1+dy*(j-1):dy*j),year2(1+dy*(j-1):dy*j),Haz2(1+dy*(j-1):dy*j),'b-')
        hold on
end

%Con lineas y puntos observados (suavizada sin elección del usuario)
figure
plot3(age(1:da),year(1:da),Hazard(1:da),'k-')
hold on
plot3(age2(1:dy),year2(1:dy),Hazard2(1:dy),'k-')
hold on
for i=2:dy
        plot3(age(1+da*(i-1):da*i),year(1+da*(i-1):da*i),Hazard(1+da*(i-1):da*i),'k-')
        hold on
        plot3(age(1+da*(i-1):da*i),year(1+da*(i-1):da*i),eta(1+da*(i-1):da*i),'k.')
        hold on
end
for j=2:da
    plot3(age2(1+dy*(j-1):dy*j),year2(1+dy*(j-1):dy*j),Hazard2(1+dy*(j-1):dy*j),'k-')
        hold on
end

%Con lineas y puntos observados (ambas)
figure
plot3(age(1:da),year(1:da),Hazard(1:da),'k-')
hold on
plot3(age2(1:dy),year2(1:dy),Hazard2(1:dy),'k-')
hold on
plot3(age(1:da),year(1:da),Haz(1:da),'b-')
hold on
plot3(age2(1:dy),year2(1:dy),Haz2(1:dy),'b-')
hold on
for i=2:dy
        plot3(age(1+da*(i-1):da*i),year(1+da*(i-1):da*i),Hazard(1+da*(i-1):da*i),'k-')
        hold on
        plot3(age(1+da*(i-1):da*i),year(1+da*(i-1):da*i),eta(1+da*(i-1):da*i),'k.')
        hold on
end
for j=2:da
    plot3(age2(1+dy*(j-1):dy*j),year2(1+dy*(j-1):dy*j),Hazard2(1+dy*(j-1):dy*j),'k-')
        hold on
end
for i=2:dy
        plot3(age(1+da*(i-1):da*i),year(1+da*(i-1):da*i),Haz(1+da*(i-1):da*i),'b-')
        hold on
end
for j=2:da
    plot3(age2(1+dy*(j-1):dy*j),year2(1+dy*(j-1):dy*j),Haz2(1+dy*(j-1):dy*j),'b-')
        hold on
end
```

Chapter 4 Non-parametric structured graduation of mortality rates

4.1 Introduction

Population censuses, surveys and vital statistics, are susceptible to having flaws (or defects) in their records either by the presence of extraordinary events (earthquakes, floods, tornados, hurricanes, etc.) or, in general, by human errors of diverse types. As it is to be expected, such flaws have negative repercussions on demographic estimations. Particularly, the wrong recording of deaths distort – or misrepresent – the phenomenon under study, which can lead to an increase (or decrease) of its intensity and timing at a certain age, in detriment to another. This situation can affect timely decision making and policy creation, both in public and private sectors. Therefore, graduating (smoothing) data come up as an alternative to solve this problem.

On the other hand, from an actuarial point of view, mortality plays a fundamental role for insurance companies, where it is important to estimate premium costs based on the risks taken. So, the predicted probability of dying must be sufficiently accurate so as to guarantee that, in the event of death, the amount of money to be paid to the insured party will be enough. Usually, data graduation is required to fulfil this requirement.

In this work, we propose a methodology to estimate mortality trends that combines mortality demographic structure with fidelity to the original data and smoothness, in such a way that the user is able to control both a smoothness percentage and a structure percentage. We emphasize that by applying this procedure, the user will be able to obtain comparable estimated trends.

This chapter is organized as follows. In Section 2, we present several non-parametric models. These models have appeared in the related literature and used to model mortality. Section 3 cites some demographic techniques in current use to project mortality. Section 4 deals with our methodological proposal, in which a signal-plus-noise mortality model is considered, together with two additional equations: one that allows inducing smoothness and another one to consider demographic structure. We introduce some smoothness and structure indexes in Section 5, where we also indicate how to use them in order to choose their associated smoothness and structure parameters. In the last section, we illustrate the practical use of our proposed methodology by way of some applications to some observed mortality data.

4.2 Non-parametric models

Haberman and Renshaw (1996) define graduation as the group of principles and methods by which observed probabilities are smoothed in order to carry out actuarial inferences and calculations. Graduation of mortality data can be done by means of parametric or non-parametric methods. In the first group, the objective is to fit a parametric function to the probabilities obtained directly form the observed data. In the second group, the actual data corresponding to death probabilities are smoothed by way of smoothing techniques. The latter methods are more flexible and appropriate to use when graduation through parametric methods is difficult. It is in such a context where we suggest using our proposal, as did Guerrero (2008).

The underlying idea of graduation and smoothing techniques is to reduce variability and facilitate the analysis of the observed data. To do so, the data are modified and turned into estimates, once unwanted fluctuations are excluded. One of the most used techniques to perform this task is the Whittaker-Henderson method, which results from minimizing the following function, for a given value of the constant $\lambda>0$,

$$(\boldsymbol{v}-\boldsymbol{u})' W(\boldsymbol{v}-\boldsymbol{u})+\lambda \boldsymbol{v}' K_d' K_d \boldsymbol{v}$$

where $\boldsymbol{u} = (u_1, \ldots, u_n)'$ is the vector of observed values, $\boldsymbol{v} = (v_1, \ldots, v_n)'$ is the vector of graduated values we are looking for, $W = diag(w_1,\ldots,w_n)$ is a weighting matrix and K_d is an $(n-d)\times n$

difference matrix, whose ij-element is given by $K_d(i,j) = (-1)^{d+i-j} d!/[(j-i)!(d-j+i)!]$ for $i = 1,...,n-d$ and $j = 1,...,n$, with $K_d(i,j) = 0$ for $j < i$ or $j > d+i$.

In the context of mortality rates, Guerrero *et al.* (2001) found that the best linear unbiased estimator of the smooth rates is Whittaker and Henderson's solution to the graduation problem. In an economic context, on the other hand, the Whittaker and Henderson's method with d = 2 is known as the Hodrick and Prescott (HP) Filter (see Hodrick and Prescott, 1997) and it is used to estimate trends in order to perform economic cycle analysis. The HP filter produces an estimate of the unobserved variable through the solution of the minimization problem

$$\min_{Y_t^*} \sum \frac{1}{\sigma_0^2}(Y_t - Y_t^*)^2 + \frac{1}{\sigma_1^2}(\nabla^2 Y_t^*)^2$$

where Y_t is the observed variable, Y_t^* is the (unobserved) trend value to be estimated, σ_0^2 is the variance of the cycle component, $\{Y_t - Y_t^*\}$, and σ_1^2 is the variance of the trend growth rate. The parameter $\lambda = \sigma_0^2 / \sigma_1^2$ serves to establish a balance between smoothness of the trend and its fidelity to the observed data.

Laxton and Tetlow (1992) proposed an extension to the HP filter. They developed the Hodrick-Prescott Multi-Variate (HPMV) filter as a tool to estimate unobserved variables including relevant economic information, as well as smoothness. Thus, the corresponding filter is obtained by minimizing a function that takes into account the random errors from one or more economic relations involving unobserved variables. That is, the HPMV filter is used to estimate the unobserved variable Y_t^* by solving the problem

$$\min_{Y_t^*} \sum (Y_t - Y_t^*)^2 + \lambda_1 (\nabla^2 Y_t^*)^2 + \lambda_2 \xi_t$$

for given values of λ_1 and λ_2. Note that this expression is similar to the one that produces the HP filter, but now it is extended with the errors (ξ_t) associated with the estimation of a given economic relation (Boone, 2000).

4.3 Mortality forecasting: demographic techniques

The Component Method (Cohort-Component Method) is the most frequently employed method to do demographic projections, both at the national level and for smaller geographic units. This method has had some light changes since its initial proposal, but its essence is still preserved. In general terms, the method is used to study the future behaviour of demographic components separately: fertility, mortality and migration, within a determined horizon (Georges et al. 2004).

To forecast mortality, the Component Method has different alternatives that allow making assumptions regarding the behaviour of mortality rates or other linked indicators. These assumptions can be grouped in: a) extrapolation techniques; b) techniques in which the mortality of an area or population is presumed in others; and c) structural models that consider changes in mortality rates due to changes in socioeconomic variables. For a) and b), some possibilities include the use of Auto-Regressive Integrated Moving Average (ARIMA) models as in Lee and Carter (1992); parametric models such as Makeham, Gompertz, and Helligman and Pollard laws, among others. Similarly, life tables from world areas can be used as basis, among them model tables that present different mortality levels and structures; the logit function, and so on. The first three options serve to interpolate death probabilities between the initial and final life tables chosen. In the last option, the initial logit life table varies linearly in time, tending towards the final logit life table.

Other methods pertaining to categories a) and b) have their support in limit mortality tables; that is, they use the lowest achieved – or almost achieved – levels, to interpolate the intermediate tables. The first proposal of limit mortality tables was presented by Bourgeois-Pichat (1952), where it was supposed that the limit levels will be reached in the long run. Those levels are the result of extrapolating mortality trends of countries with high life expectancy. The hypothesis underlying this kind of method supposes that mortality will evolve depending on the level and structure of deaths, according to the world region it belongs to. This argument is supported by Demographic Transition Theory. Regarding human survival limits, the works of Olshanky *et al.* (1990, 2001) and Oeppen and Vaupel (2002) are interesting because, for instance, they studied the reductions in mortality that are required to achieve a life expectancy at birth that grows from 80 to 120 years and its influence on different areas of public policy.

In case b), for example, the goal technique was used. Such a technique is based on the idea that for a given population, mortality rates will converge towards those observed in another *goal population*. Such a population is chosen in such a way that it provides a set of believable goals to be reached by the population projected. The choice of a goal population is based on similarities regarding cultural and socioeconomic characteristics, medical advances and first causes of mortality (Olshansky, 1988). An alternative way to present the goals is by means of the so called *cause delay*. With such an approach, the goal population is a young cohort of the same population instead of the same cohort in a different population. The focus is on the implications of delaying or fully eliminating the occurrence of one or more causes of mortality (Manton *et al.* 1980; Olshansky, 1987). The basic premise behind the method is that changes in life styles and medical advances, delay the occurrence of several causes of mortality until advanced ages. Therefore, each cohort has lower risk of dying than the previous cohorts.

4.4 Proposed methodology

We suggest using the HPMV filter to estimate mortality trends by incorporating the idea of data smoothness. To that end, a signal-plus-noise model is presented at first,

$$Y_t = Y_t^S + \eta_t$$

where Y_t denotes the observed mortality, Y_t^S is the signal, which in our case represents the smooth mortality trend and η_t is the noise that basically obscures the trend. When penalizing for the lack of smoothness and minimizing with respect to Y_t^S, the following problem arises

$$\min_{Y_t^S} \sum_{t=1}^{n} (Y_t - Y_t^S)^2 + \lambda_1 (\nabla^d Y_t^S)^2 + \lambda_2 \delta_t,$$

with δ_t the random error of a structural demographic model. Then, we have a problem similar to Boone's (2000), where we now intend to estimate the unobserved mortality trend, Y_t^S, by solving the aforementioned minimization problem.

We approach this problem by first defining a smoothness index which helps to choose the constants λ_1 and λ_2. It is important to note that the methodology proposed is interpreted according to a demographic theory that allows for valid comparisons between mortality trends. Thus, we consider the model

$$\boldsymbol{Y} = \boldsymbol{Y}^S + \boldsymbol{\eta}, \quad \boldsymbol{\eta} \sim (\boldsymbol{0},\ \sigma_\eta^2 I_n) \tag{1}$$

$$K_2\boldsymbol{Y}^S = \boldsymbol{\varepsilon}, \quad \boldsymbol{\varepsilon} \sim (\boldsymbol{0},\ \sigma_\varepsilon^2 I_{n-2}), \quad \mathrm{E}(\boldsymbol{\varepsilon\eta}') = 0 \tag{2}$$

and

$$\boldsymbol{U} = \boldsymbol{Y}^S + \boldsymbol{\delta}, \quad \boldsymbol{\delta} \sim (\boldsymbol{0},\ \sigma_\delta^2 I_n), \quad \mathrm{E}(\boldsymbol{\delta\eta}') = 0, \quad \mathrm{E}(\boldsymbol{\delta\varepsilon}') = 0. \tag{3}$$

where the symbol ~ stands for "distributed as" (mean vector, variance-covariance matrix).

Equation (1) expresses the vector of mortality as a trend vector $\boldsymbol{Y}^S$ plus a random noise vector $\boldsymbol{\eta}$, with σ_η^2 being the noise variance and I_n the n-dimensional identity matrix. In (2) we have an equation that induces smoothness in the behaviour of $\boldsymbol{Y}^S$ by assuming an underlying polynomial of degree one, that is, $Y_t^S = 2Y_{t-1}^S + Y_{t-2}^S + \varepsilon_t$ for $t = 3, ..., n$, where ε_t is a random error with variance σ_ε^2. And finally, in (3) we postulate a mortality experience with limit structure or, seen differently, we use another source of data to combine with the observed information.

We can write (1)-(3) as the following system of equations

$$\begin{pmatrix} \boldsymbol{Y} \\ \boldsymbol{0} \\ \boldsymbol{U} \end{pmatrix} = \begin{pmatrix} I_n \\ K_2 \\ I_n \end{pmatrix} \boldsymbol{Y}^S + \begin{pmatrix} \boldsymbol{\eta} \\ -\boldsymbol{\varepsilon} \\ \boldsymbol{\delta} \end{pmatrix}, \text{ with } \begin{pmatrix} \boldsymbol{\eta} \\ -\boldsymbol{\varepsilon} \\ \boldsymbol{\delta} \end{pmatrix} \sim \left(\begin{pmatrix} \mathbf{0} \\ \mathbf{0} \\ \mathbf{0} \end{pmatrix}, \Sigma \right) \text{ where } \Sigma = \begin{pmatrix} \sigma_\eta^2 I_n & 0 & 0 \\ 0 & \sigma_\varepsilon^2 I_{n-2} & 0 \\ 0 & 0 & \sigma_\delta^2 I_n \end{pmatrix}.$$

thus, by using Generalized Least Squares (GLS) to estimate $\boldsymbol{Y}^S$, we have

$$\hat{\boldsymbol{Y}}^S = \left[\begin{pmatrix} I_n \\ K_2 \\ I_n \end{pmatrix}' \Sigma^{-1} \begin{pmatrix} I_n \\ K_2 \\ I_n \end{pmatrix} \right]^{-1} \begin{pmatrix} \mathrm{I_n} \\ \mathrm{K_2} \\ \mathrm{I_n} \end{pmatrix}' \Sigma^{-1} \begin{pmatrix} \boldsymbol{Y} \\ \boldsymbol{0} \\ \boldsymbol{U} \end{pmatrix}$$

$$= (\sigma_\eta^{-2} I_n + \sigma_\varepsilon^{-2} K'_2 K_2 + \sigma_\delta^{-2} I_n)^{-1} (\sigma_\eta^{-2} \boldsymbol{Y} + \sigma_\delta^{-2} I_n \boldsymbol{U}). \tag{4}$$

Then, if we let $\lambda_1 = \sigma_\eta^2 / \sigma_\varepsilon^2$ and $\lambda_2 = \sigma_\eta^2 / \sigma_\delta^2$, we obtain

$$\hat{\boldsymbol{Y}}^S = (I_n + \lambda_1 K'_2 K_2 + \lambda_2 I_n)^{-1} (\boldsymbol{Y} + \lambda_2 \boldsymbol{U}) \tag{5}$$

whose variance-covariance matrix is given by

$$\Gamma = \mathrm{Var}(\hat{\boldsymbol{Y}}^S) = (I_n + \lambda_1 K'_2 K_2 + \lambda_2 I_n)^{-1} \sigma_\eta^2. \tag{6}$$

Hence, we have $\hat{\boldsymbol{Y}}^S = \mathrm{M}(\boldsymbol{Y} + \lambda_2 \boldsymbol{U})$ and $\Gamma = \mathrm{M}\sigma_\eta^2$ with $\mathrm{M} = (I_n + \lambda_1 K'_2 K_2 + \lambda_2 I_n)^{-1}$.

Similarly, if we write $M = (I_n + \dfrac{\lambda_1}{1+\lambda_2} K'_2 K_2)^{-1} (1+\lambda_2)^{-1}$, equation (5) can be rewritten as

$$\hat{\mathbf{Y}}^{\mathrm{S}} = (I_n + \alpha\lambda_1 K'_2 K_2)^{-1} (\alpha \boldsymbol{Y} + (1-\alpha)\boldsymbol{U}), \text{ with } \alpha = (1+\lambda_2)^{-1}. \tag{7}$$

From here, it can be seen that $\hat{\boldsymbol{Y}}^S \to \boldsymbol{U}$, if $\alpha \to 0$. Therefore, the smoothness induced by (2) disappears and only the convergence to the structure given by (3) is taken into account. On the other hand, if $\alpha \to 1$, $\hat{\boldsymbol{Y}}^S \to (I_n + \lambda_1 K'_2 K_2)^{-1}\boldsymbol{Y}$, and the usual HP filter is obtained. Notice that the value of α must be known in advance to calculate $\hat{\boldsymbol{Y}}^S$. Besides, $\hat{\boldsymbol{Y}}^S$ can be interpreted as the combination of two sources of information, the weight of which can be decided by the analyst when choosing a value for the constant α.

From a numerical calculation standpoint, the smoothed vector (7) can be conveniently obtained by applying the Kalman Filter (KF) with smoothness. In order to apply this filter we make use of models (1) and (3), so that

$$Y_t = Y_t^S + \eta_t,\ \ U_t = Y_t^S + \delta_t,\ \ \eta_t \sim (0, \sigma_\eta^2),\ \ \delta_t \sim (0, \sigma_\delta^2),\ \ E(\eta_t \varepsilon_t) = 0,$$

imply

$$\begin{aligned}\alpha Y_t + (1-\alpha)U_t &= \alpha Y_t^S + \alpha\eta_t + (1-\alpha)Y_t^S + (1-\alpha)\delta_t \\ &= Y_t^S + \gamma_t\end{aligned}$$

with $\gamma_t = \alpha\eta_t + (1-\alpha)\delta_t \sim (0, \sigma_\gamma^2)$ and $\sigma_\gamma^2 = \alpha^2\sigma_\eta^2 + (1-\alpha)^2\sigma_\delta^2$. Therefore, a state-space model can be expressed with the following measuring and transition equations, respectively

$$\alpha Y_t + (1-\alpha)U_t = \boldsymbol{c}_t'\boldsymbol{X}_t + \gamma_t$$

and

$$\boldsymbol{X}_t = A_t \boldsymbol{X}_{t-1} + \boldsymbol{w}_t$$

where

$$\boldsymbol{X}_t = \begin{pmatrix} Y_t^S \\ Y_{t-1}^S \end{pmatrix},\ A_t = \begin{pmatrix} 2 & -1 \\ 1 & 0 \end{pmatrix},\ \boldsymbol{c}_t = \begin{pmatrix} 1 \\ 0 \end{pmatrix} \text{ and } \boldsymbol{w}_t = \begin{pmatrix} \varepsilon_t \\ 0 \end{pmatrix}.$$

So, the KF can be used as in Guerrero (2008), but instead of using the original data Y_t we will now use $\alpha Y_t + (1-\alpha)U_t$, with a known α value.

It is desirable to know the existing relationship between the uncertainties of Y_t and $\alpha Y_t + (1-\alpha)U_t$. Thus, we consider the variance σ_γ^2 of the new variable $\alpha Y_t + (1-\alpha)U_t$ which is given by

$$\sigma_\gamma^2 = \alpha^2\sigma_\eta^2 + (1-\alpha)^2\sigma_\delta^2 \text{ with } \alpha = (1+\lambda_2)^{-1} \text{ and } \lambda_2 = \sigma_\eta^2 / \sigma_\delta^2$$

so that

$$\alpha = \frac{\sigma_\delta^2}{\sigma_\delta^2 + \sigma_\eta^2} \text{ and } 1-\alpha = \frac{\sigma_\eta^2}{\sigma_\delta^2 + \sigma_\eta^2},$$

from which we get

$$\alpha^2\sigma_\eta^2 + (1-\alpha)^2\sigma_\delta^2 = \left(\frac{\sigma_\delta^2}{\sigma_\delta^2 + \sigma_\eta^2}\right)^2 \sigma_\eta^2 + \left(\frac{\sigma_\eta^2}{\sigma_\delta^2 + \sigma_\eta^2}\right)^2 \sigma_\delta^2 .$$

That is,

$$\sigma_\gamma^2 = \alpha\,\sigma_\eta^2,$$

where $0 < \alpha < 1$ and σ_η^2 is the variance of the model for Y_t. Hence, we conclude that there is more uncertainty in the behaviour of Y_t than in that of $\alpha Y_t + (1-\alpha)U_t$, once α is known.

4.5 Smoothness index and its use to select the smoothness parameters

To measure the proportion of relative precision $\sigma_\eta^{-2} I_n$ in relation to the total precision attained through the estimation process, $\sigma_\eta^{-2} I_n + \sigma_\delta^{-2} I_n + \sigma_\varepsilon^{-2} K'_2 K_2$, we propose to use the index

$$\Lambda(\sigma_\eta^{-2} I_n; \sigma_\eta^{-2} I_n + \sigma_\delta^{-2} I_n + \sigma_\varepsilon^{-2} K'_2 K_2) = tr[\sigma_\eta^{-2} I_n (\sigma_\eta^{-2} I_n + \sigma_\delta^{-2} I_n + \sigma_\varepsilon^{-2} K'_2 K_2)^{-1}]/n \quad (8)$$

where tr(.) denotes trace of a matrix, while $\sigma_\eta^{-2} I_n$, $\sigma_\delta^{-2} I_n$ and $\sigma_\varepsilon^{-2} K'_2 K_2$ are $n \times n$ positive definite matrices. This index is a measure of relative precision that satisfies the following properties: (i) it takes on values between zero and one; (ii) it is invariant under linear transformations of the variable Y involved; (iii) it behaves linearly; and (iv) it adds up to unity, *i. e.*

$$\Lambda(\sigma_\eta^{-2} I_n; \sigma_\eta^{-2} I_n + \sigma_\delta^{-2} I_n + \sigma_\varepsilon^{-2} K'_2 K_2) + \Lambda(\sigma_\delta^{-2} I_n; \sigma_\eta^{-2} I_n + \sigma_\delta^{-2} I_n + \sigma_\varepsilon^{-2} K'_2 K_2)$$
$$+ \Lambda(\sigma_\varepsilon^{-2} K'_2 K_2; \sigma_\eta^{-2} I_n + \sigma_\delta^{-2} I_n + \sigma_\varepsilon^{-2} K'_2 K_2) = 1$$

The proof that Λ is the unique scalar measure fulfilling the four criteria follows directly from the proof provided by Theil (1963) for the case of two positive definite matrices A and B, where the index is given by $\Lambda(A; A+B)$. We only need to recognize that, for instance, our $\sigma_\eta^{-2} I_n$ plays the role of A and $\sigma_\delta^{-2} I_n + \sigma_\varepsilon^{-2} K'_2 K_2$ plays that of B.

This index is useful to quantify the relative precision attributable to smoothness and to the induced structure in the model, which are part of the precision matrix Γ^{-1} given by (6). Therefore, we define the smoothness index

$$S(\lambda_1, \lambda_2; n) = tr[\sigma_\varepsilon^{-2} K'_2 K_2 (\sigma_\eta^{-2} I_n + \sigma_\varepsilon^{-2} K'_2 K_2 + \sigma_\delta^{-2} I_n)^{-1}]/n$$
$$= 1 - tr\left\{[I_n + \lambda K'_2 K_2]^{-1}\right\}/n$$

with

$$\lambda = (\sigma_\eta^{-2} + \sigma_\delta^{-2})^{-1} \sigma_\varepsilon^{-2} = \lambda_1 (1+\lambda_2)^{-1} = \alpha\lambda_1 .$$

Since λ is associated with the smoothness of $\alpha \boldsymbol{Y} + (1-\alpha)\boldsymbol{U}$, its value can be chosen with the aid of the smoothness index S(λ_1, λ_2; *n*). On the other hand, λ_1 corresponds to the smoothness parameter of the original data $\boldsymbol{Y}$ and it can be deduced from the values of λ and α; that is, $\lambda_1 = \lambda / \alpha$ with $\alpha > 0$. In the same fashion, if λ_1 is first set as the smoothness parameter leading to a desired percentage of smoothness for $\boldsymbol{Y}$, and if $\alpha \in (0,1]$ is set later, we can deduce the value of λ that determines the smoothness of $\alpha \boldsymbol{Y} + (1-\alpha)\boldsymbol{U}$. The previous ideas could be used to first set λ_1, when choosing the percentage of smoothness for ***Y***, then set $\lambda = \alpha\lambda_1$, when choosing the percentage of smoothness for the combination $\alpha \boldsymbol{Y} + (1-\alpha)\boldsymbol{U}$.

Notice that the percentage of smoothness for ***Y*** should be greater than, or equal to that of the combination, because $\lambda = \alpha\lambda_1 \leq \lambda_1$, since $0 < \alpha \leq 1$ and the smoothness index is an increasing monotone function. Or else, the value of α could be set according to what was previously said by setting the values of λ_1 and λ, based on the smoothness index S(λ_1; *n*) = $1 - tr\left\{[I_n + \lambda_1 K'_2 K_2]^{-1}\right\}/n$, applicable to $\alpha \boldsymbol{Y} + (1-\alpha)\boldsymbol{U}$. This index is associated to the smoothness of ***Y*** alone, which corresponds to α =1. In this case, λ_2 = 0 and the estimate becomes

$$\hat{\boldsymbol{Y}}^S = (I_n + \lambda_1 K'_2 K_2)^{-1} \boldsymbol{Y}$$

with Var($\hat{\boldsymbol{Y}}^S$) = $(I_n + \lambda_1 K'_2 K_2)^{-1} \sigma_\eta^{-2}$. Of course, the solution that includes both smoothness and demographic structure corresponds to $\alpha \in (0,1)$, that is to say, when λ_2 > 0, which is provided by (7).

In short, the strategy to smooth the dataset $\{Y_1,...,Y_N\}$ with the HPMV filter, using known structural data $\{U_1,...,U_N\}$ consists of the following steps:

1. Smooth the $\boldsymbol{Y}$ data without considering the existence of ***U***. Thus, fix a desired percentage of smoothness and apply Guerrero's (2008) procedure. As a result, the value of λ_1 is deduced and the corresponding smoothed curve with 100S(λ_1; n)% of smoothness (for example, 80%) is obtained.

2. Decide the degree of smoothness to be exchanged with structure, so that the percentage of smoothness is reduced (let us say from 80% to 75%). By doing so, fix the value of 100S(λ_1, λ_2; n)% and deduce the corresponding value of $\alpha \in (0,1)$ from it.

3. Perform the smoothing process with structure by applying the KF to the data $\{\alpha Y_t + (1-\alpha)U_t\}$ which will result in 100S(λ_1, λ_2; *n*)% smoothness and 100[S(λ_1; *n*) -S(λ_1, λ_2; *n*)]% structure (that is, proximity to ***U***).

It should be realized that $tr[(I_n + \lambda K'_d K_d)^{-1}] \rightarrow d$ when $\lambda \rightarrow \infty$, where d is the order of the K_d difference matrix (Eilers and Marx, 1996: 94). Therefore, the maximum smoothness that can be obtained with n observations is S(λ; *n*) → 1-*d/n* when $\lambda \rightarrow \infty$. This result is useful to know in advance the maximum percentage of smoothness achievable in practical applications.

4.6 Applications

Both approaches of the methodology proposed are used in what follows. First, we show with two examples how to use our method to obtain structure and smoothness in a proper way; then, we present two other examples in which the analyst has the opportunity to decide which source of information has greater credibility. All calculations were carried out with the aid of the computer package RATS 7.0 (see Appendix 4.8.1-4.8.4).

The data employed in these illustrations come from different information sources. The crude forces of mortality of the United Kingdom, Japan and Chile, as well as the United States death probabilities estimated by period, are taken from http://www.mortality.org/. United States death probabilities (q_x) estimated by cohorts are taken from Life Tables for the Social Security Area by Calendar Year (www.ssa.gov/OACT/NOTES/as120/LifeTables_Tbl_7.html). For the Mexico City example, data come from a comparative analysis between paleodemography and historical demography for the XIX Century (Ortega, 2003). Natural logarithms were used in all cases.

In the first example, we propose a 2010 goal, in such a way that the year 2000 male population in the United Kingdom has a mortality experience as the one in Japan in 2006. We have

$N = 101$ data points, so that the maximum smoothness achievable is 98.02%. With a chosen initial smoothness of 75% and final one of 70%, we obtained $\lambda_I = 6$ and $\lambda = 3$, so that $\alpha = 0.5$. Notice how the estimated trend gives greater weight to Japanese mortality in almost all life range, except for mortality in children under 1 year of age, where it is slightly below the observed data.

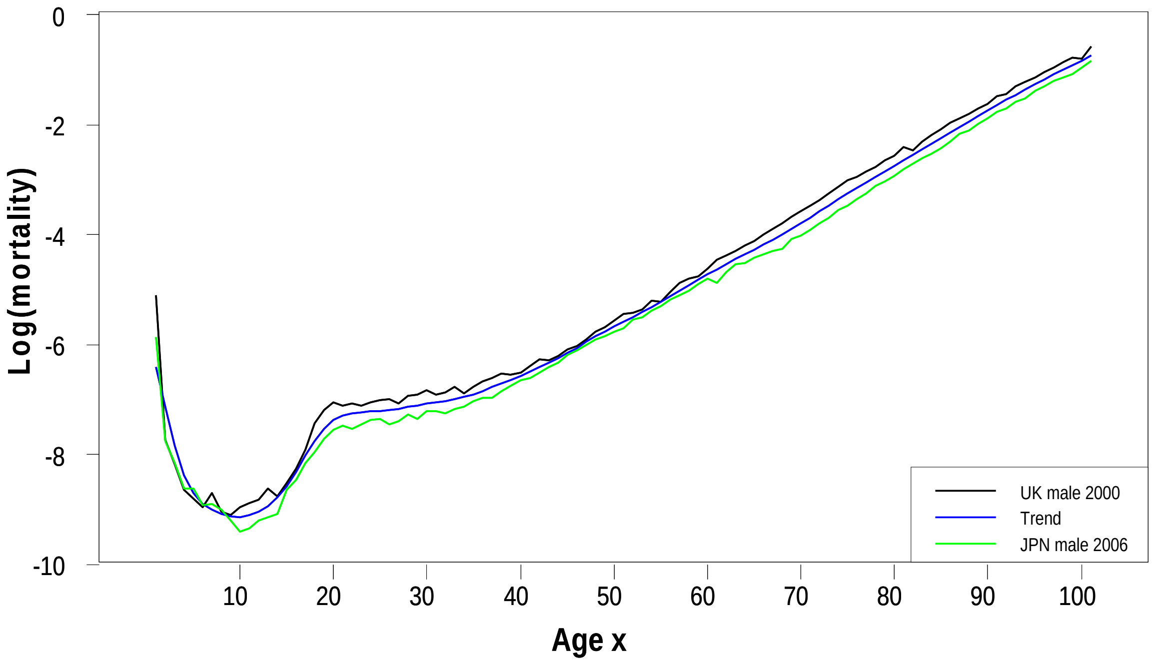


Figure 4.1. Male log(mortality) observed in UK 2000, Japan 2006 and trend with 70% smoothness ($\lambda = 3$ and $\alpha = 0.5$).

For the second example, we use a Chile's female population goal for 2010, such that the annual mortality indicator is to have the same experience as the Japanese women for 2006. For this case, we also have $N = 101$ and the same values for λ_I, λ and α, as well as the chosen initial and final smoothness. Also, according to Figure 4.2, the estimated trend balances when both experiences move apart from each other in specific segments of the life range. On the other hand, the estimated child mortality for children less than 1 year of age is very close, as much as one from another experience.

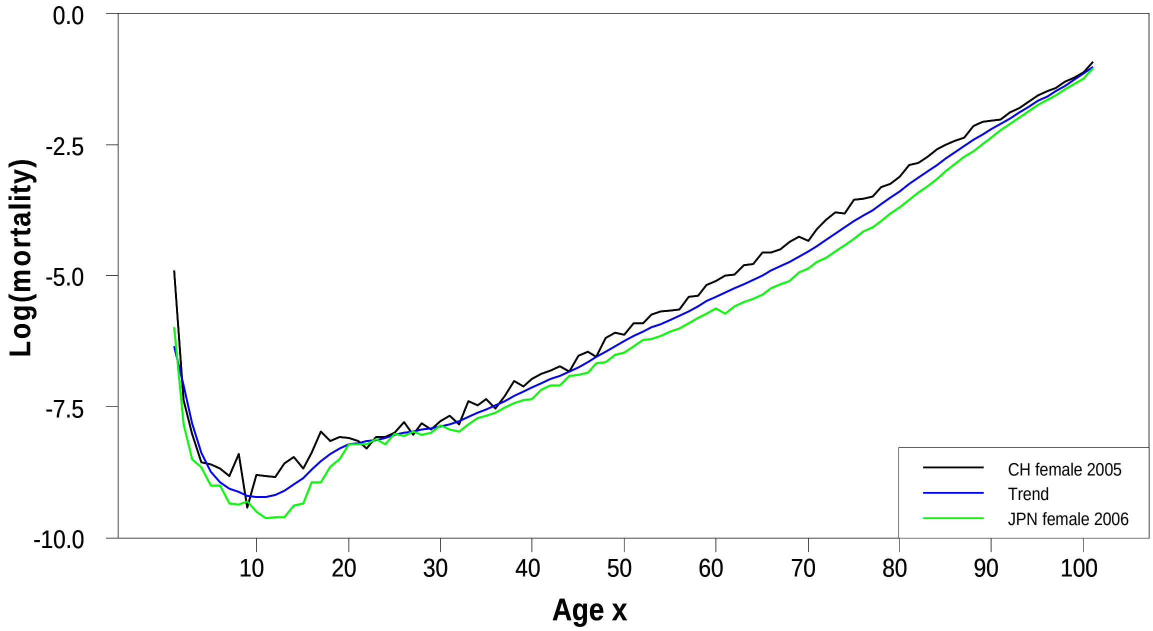


Figure 4.2. Female log(mortality) observed in Chile 2005, Japan 2006 and trend with 70% smoothness ($\lambda = 3$ and $\alpha = 0.5$).

In the last two examples – in contrast with the first two – the sample sizes of the original mortality series are different, a situation that does not cause any problem, since we estimate trends using KF (when missing data appear, the filter is applied without smoothness). The larger of the two series is used as the Y series of the model. So, two information sources are used, and the analyst can grant greater, equal, or less credibility to one of them when choosing a specific value for the parameter (when $\alpha = 0.5$, the same credibility is given to both sources). It is important to point out that, with this approach, the observed mortality structure does not necessarily aspires to behave as another one, but the analyst wants to merge two sources of information into one and has to decide how to weight them in a linear combination.

The third example makes use of the United States mortality for the male population, as seen from a longitudinal (by cohort) approach and by period. The corresponding years are 2010 and 2000, respectively. In this case, the series have 120 and 110 data points and the maximum smoothness achievable – based on the (largest) longitudinal series – is 98.33%. The chosen initial smoothness is 90% and the parameter values become λ_1 = 14, λ = 8.4 and α = 0.6, generating a final smoothness of 77.6%. In Figure 4.3, it can be noticed that the resulting estimated trend is balanced between both experiences. Regarding mortality for males under 15 years of age, the trend is more similar to the longitudinal experience and is greater than the one reported by period.

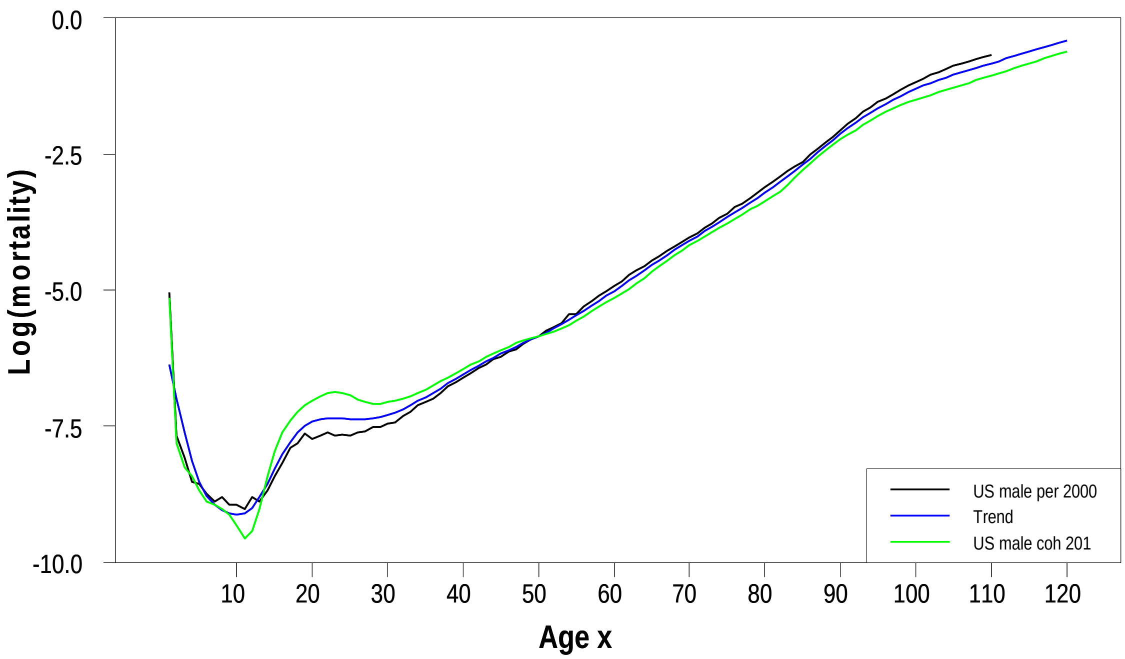


Figure 4.3. Log(mortality) observed by period 2000 and cohort 2010 for the US and trend with 77.6% smoothness ($\lambda = 8.4$ and $\alpha = 0.6$).

The last example shows how this methodology can be used by some specialists, such as anthropologists, demographers or statisticians. In fact, starting from a paleodemography or a historical demography approach, it is feasible to obtain mortality trend estimates. Let ${}_nq_x$ be the death probabilities by quinquennial groups corresponding to the XIX Century, that come from Santa Paula cemetery and Santa Maria parish, both located in Mexico City. There are N = 19 observations for the Parish series and 13 for the Cemetery series. The maximum smoothness achievable in the longer series is 89.47%, so initial smoothness is set at 80% while final smoothness became 79.1% with the choice of α = 0.8. The parameter values employed are λ_1 = 35 and λ = 28. As with the previous example, and despite the Cemetery series has the highest variability, the estimated trend is balanced between both sources. However, the choice of α, based on the analyst's knowledge, is of paramount importance for this purpose.

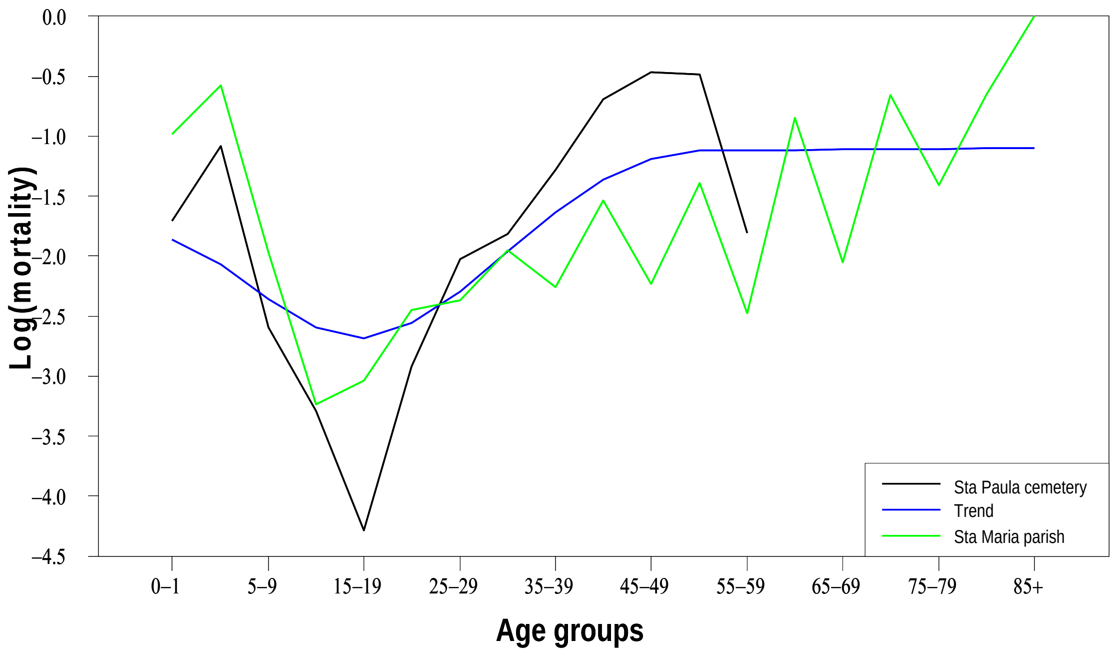


Figure 4.4. Log(mortality) observed in the XIX Century Mexico City and trend with 77.6% smoothness ($\lambda = 35$ and $\alpha = 0.8$).

4.7 Conclusions

The methodology proposed in this work is useful to estimate trends in mortality series when considering fit, smoothness and some additional information coming from a mortality structure. Also, it allows the analyst to control smoothness and structure percentages according to his/her interests in order to achieve comparability. The analyst can decide which approach of the two possible applications to use in a specific situation. The flexibility to handle different mortality indicators was illustrated empirically with some observed mortality experiences. These illustrations suggest the feasibility of applying our proposal in different scientific fields, not only in demography.

Some of the circumstances that could come up when applying this proposal are: (i) the presence of missing data and (ii) different size information sources. Both cases can be handled very efficiently by using the KF as a computing device, which easily overcomes the possible numeric difficulties that may arise, for instance, when inverting matrices. Finally, it is worth mentioning that the application of the proposed methodology can be done on other kind of demographic indicators, such as fertility, marriage, divorces, and migration.

As a future work, we intend to generalize this methodology to the two-dimensional case, where it is foreseen that (as it happened with the one-dimensional case) there could be interesting theoretical results in which the different smoothness parameters are related. Then, it would be appropriate to apply this technique to generate mortality surface estimates, restricted to the experience and values that the analyst considers appropriate, in order to graduate information and enhance comparability.

Another future work will consider the practical question of applying the methodology by chunks of the series within the age ranges could arise, both for the one and the two-dimensional cases. This need appears when the analyst wants a closer proximity with a demographic structure in a specific range and keeps the rest, for example, in a balanced way between the different sources of information. One of the most remarkable advantages of the methodology is the possibility for the analyst to give greater credibility to an information source over the other.

4.8 Appendix

4.8.1 RATS routine: Male log(mortality) observed in UK 2000, Japan 2006 and trend

```
* LOS CALCULOS CORRESPONDEN AL FILTRO DE HODRICK Y PRESCOTT Y SE
* APLICAN MEDIANTE LA IDEA DEL NUEVO ARTICULO Y EL FILTRO DE KALMAN CON SUAVIZAMIENTO.
*
*********************************************************************
*
* AQUI SE ESPECIFICA LA LONGITUD DE LA SERIE DE MORTALIDAD.
*
DECLARE REAL N1
COMPUTE INI = 1                              ;*DATO REQ.
COMPUTE FIN = 101                             ;*DATO REQ.
COMPUTE N = FIN-INI+1
COMPUTE N1= FIN-INI+1
DECLARE REAL SUAF%

OPEN DATA C:\3_ST\Finales\dataUKM.xls                   ;*ARCH. REQ.
DATA(FORMAT=xls,ORG=OBS) / UKM

OPEN DATA C:\3_ST\Finales\dataJPM.xls                   ;*ARCH. REQ.
DATA(FORMAT=xls,ORG=OBS) / JPM

*open styles thicklines.txt
*grparm(import=styles)

LABELS UKM JPM ;*Etiquetas para las variables DATOS y TENDENCIA
# 'UK MALE 2000' 'JPN MALE 2006'
GRAPH(Style=Line,$
key=lorigth,vlabel='Log(qx)',hlabel='Age x') 2
# UKM
# JPM

DECLARE RECTANGULAR IN K2
DIM IN(N,N)
COMPUTE IN=%identity(N)
COMPUTE K2=%zeros(N-2,N)
DO J=1,N
         DO I=1,N-2
   COMPUTE K2(I,I)=1
                  COMPUTE K2(I,I+1)=-2
   COMPUTE K2(I,I+2)=1
  END DO
END DO

************************
********STEP 1***********
**ESTIMACION DE LAMBDA1**

DECLARE REAL LAMBDA1
DECLARE REAL NMIN
DECLARE REAL MAXS

COMPUTE SUAF% = 75 ;*suavidad final deseada por el usuario
COMPUTE NMIN = 2/(1-(SUAF%/100)) ;*número mínimo de observaciones para aspirar a la suavidad deseada
COMPUTE MAXS = 1-(2/N1)    ;*máxima suavidad que se puede alcanzar

DISPLAY '               '
DISPLAY '         STEP 1'
DISPLAY 'N MIN            ' NMIN
```

```
DISPLAY 'MAX SMOOTHNESS    ' MAXS

COMPUTE LAMBDA1 = 1
COMPUTE PARC = 1-(%trace(inv(%identity(N)+1*tr(K2)*K2))/N)
COMPUTE SUAF = SUAF%/100
DISPLAY 'INITIAL SMOOTHNESS' PARC
DISPLAY 'DESIRED SMOOTHNESS' SUAF

WHILE PARC<SUAF {
            COMPUTE LAMBDA1 = LAMBDA1+1
            COMPUTE PARC = 1-(%trace(inv(%identity(N)+LAMBDA1*tr(K2)*K2))/N)
}
END WHILE

DISPLAY 'LAMBDA1          ' LAMBDA1
DISPLAY 'FINAL SMOOTHNESS  ' PARC

****METODO A TRAVES DEL FILTRO DE KALMAN
* AQUI INICIA EL CALCULO DE LA TENDENCIA CON EL FILTRO DE KALMAN.

DECLARE RECT a c                        ;*Declaracion de las matrices A y c
DECLARE FRML[VECT] y                            ;*Declaracion del vector y a ser estimado
DECLARE SYMM sw v0                              ;*Declaracion de matrices simétricas
DECLARE VECT x0 sv                      ;*Declaracion de vectores de valores iniciales
COMPUTE a = ||2.0,-1.0|1.0,0.0||        ;*Matriz de transicion de la ecuacion de estado
COMPUTE c  = ||1.0|0.0||                ;*Matriz de la ecuacion de medida
FRML y  = ||UKM||                         ;*Asignacion de los UKM observados al vector y
COMPUTE sw = ||1.0|0.0,0.0||            ;*Asignacion de la matriz simétrica de covarianzas de las w´s. Es ||1.0, 0.0|0.0,0.0||
COMPUTE sv = ||LAMBDA1||                    ;*Asignacion de la matriz simétrica de covarianzas de las v´s. Valor de sigma eta del
paper, final pag.189.
COMPUTE x0 = ||UKM(INI),UKM(INI)||                        ;*Matriz de valores iniciales
COMPUTE v0 = ||4.0*LAMBDA1|4.0*LAMBDA1,4.0*LAMBDA1||    ;*Da la covarianza de x0
DLM(a=a,c=c,y=y,sv=sv,sw=sw,x0=x0,sx0=v0,TYPE=SMOOTH) INI FIN STATES  ;*TYPE=SMOOTH se refiere al Kalman
smoother
SET TENDENCIA1 = STATES(t) (1)                            ;*Asignacion de la variable tendencia a partir del comando DLM
SET CICLO = UKM-TENDENCIA1           ;*Estimacion de la variable ciclo
LABELS UKM TENDENCIA1               ;*Etiquetas para las variables DATOS y TENDENCIA
# 'UK MALE 2000' 'TREND'

* AQUI SE GRAFICAN LOS RESULTADOS Y SE GUARDAN EN UN ARCHIVO DE EXCEL.
*
DISPLAY(STORE=SUBTIT) 'N =' ### N ', Smoothness =' ##.# SUAF% $
                  '%  y  Lambda1 =' #####.# LAMBDA1
GRAPH(Style=Line,SUBHEADER=SUBTIT,$
key=lorigth,vlabel='Log(qx)',hlabel='Age x') 2
# UKM
# TENDENCIA1
*
OPEN COPY C:\3_ST\Finales\UKM_S.XLS                       ;*ARCH. REQ.
COPY(FORMAT=XLS,ORG=OBS,DATES) / UKM TENDENCIA1

************************
*******STEP 2***********

DECLARE REAL LAMBDA
DECLARE REAL ALFA
DECLARE REAL SUAF2%
DECLARE REAL NMIN2
DECLARE REAL MAXS2

COMPUTE SUAF2% = 70 ;*suavidad final deseada por el usuario
```

```
COMPUTE NMIN2 = 2/(1-(SUAF2%/100)) ;*número mínimo de observaciones para aspirar a la suavidad deseada
COMPUTE MAXS2 = 1-(2/N1)    ;*máxima suavidad que se puede alcanzar

DISPLAY '                '
DISPLAY '          STEP 2'
DISPLAY 'N MIN           ' NMIN2
DISPLAY 'MAX SMOOTHNESS    ' MAXS2

COMPUTE LAMBDA = 1
COMPUTE PARC2 = 1-(%trace(inv(%identity(N)+1*tr(K2)*K2))/N)
COMPUTE SUAF2 = SUAF2%/100

DISPLAY 'INITIAL SMOOTHNESS' PARC2
DISPLAY 'DESIRED SMOOTHNESS' SUAF2

WHILE PARC2<SUAF2 {
            COMPUTE LAMBDA = LAMBDA+1
            COMPUTE PARC2 = 1-(%trace(inv(%identity(N)+LAMBDA*tr(K2)*K2))/N)
}
END WHILE

DISPLAY 'LAMBDA          ' LAMBDA
DISPLAY 'FINAL SMOOTHNESS  ' PARC2

COMPUTE ALFAU = 0.5
COMPUTE OPC = 1               ; *WITH OPC = 1, ALFA IS ESTIMATED
                                                                                         *WITH OPC = 2,
ALFA IS SELECTED BY THE USER
COMPUTE ALFA = %IF(OPC==1,LAMBDA/LAMBDA1,ALFAU)
COMPUTE SUAF2% = %IF(OPC==1,SUAF2%,(1-(%trace(inv(%identity(N)+LAMBDA1*ALFAU*tr(K2)*K2))/N))*100)

DISPLAY 'SELECTED OPTION           ' OPC
DISPLAY 'ALFA                               ' ALFA
DISPLAY 'SUAF                               ' SUAF2

COMPUTE LAMBDA2 =(LAMBDA1/LAMBDA)-1

DISPLAY 'LAMBDA2                            ' LAMBDA2

DECLARE VECTOR YC1(N)

DO i=1,N
COMPUTE YC1(i) = ALFA*%IF(i<=110,UKM(i),TENDENCIA1(i))+(1-ALFA)*JPM(i)
END DO i
SET YC 1 N =YC1(t)
PRINT 1 N-6 YC

* AQUI INICIA EL CALCULO DE LA TENDENCIA CON EL FILTRO DE KALMAN.

DECLARE RECT a c                    ;*Declaracion de las matrices A y c
DECLARE FRML[VECT] y                         ;*Declaracion del vector y a ser estimado
DECLARE SYMM sw v0                           ;*Declaracion de matrices simétricas
DECLARE VECT x0 sv                  ;*Declaracion de vectores de valores iniciales
COMPUTE a = ||2.0,-1.0|1.0,0.0||    ;*Matriz de transicion de la ecuacion de estado
COMPUTE c  = ||1.0|0.0||            ;*Matriz de la ecuacion de medida
FRML y  = ||YC||                      ;*Asignacion de los YC observados al vector y
COMPUTE sw = ||1.0|0.0,0.0||        ;*Asignacion de la matriz simétrica de covarianzas de las w´s. Es ||1.0, 0.0|0.0,0.0||
COMPUTE sv = ||LAMBDA||                ;*Asignacion de la matriz simétrica de covarianzas de las v´s. Valor de sigma eta del
paper, final pag.189.
COMPUTE x0 = ||YC(INI),YC(INI)||                ;*Matriz de valores iniciales
COMPUTE v0 = ||4.0*LAMBDA|4.0*LAMBDA,4.0*LAMBDA||    ;*Da la covarianza de x0
```

```
DLM(a=a,c=c,y=y,sv=sv,sw=sw,x0=x0,sx0=v0,TYPE=SMOOTH) INI FIN STATES  ;*TYPE=SMOOTH se refiere al Kalman
smoother
SET TENDENCIA2 = STATES(t) (1)                        ;*Asignacion de la variable tendencia a partir del comando DLM
SET CICLO = YC-TENDENCIA2           ;*Estimacion de la variable ciclo a partir del comando DLM

LABELS UKM TENDENCIA2 JPM           ;*Etiquetas para las variables YC y TENDENCIA
# 'UK male 2000' 'Trend' 'JPN male 2006'
DISPLAY(STORE=SUBTIT) 'N =' ### N ', Smoothness =' ##.# SUAF2% $
                '%, Lambda =' #####.# LAMBDA ', Alfa =' #.## ALFA
GRAPH(Style=Line,key=lorigth,$
vlabel='Log(qx)',hlabel='Age x',SUBHEADER=SUBTIT) 3
# UKM
# TENDENCIA2
# JPM
*
OPEN COPY C:\3_ST\Finales\YC_S.XLS                      ;*ARCH. REQ.
COPY(FORMAT=XLS,ORG=OBS,DATES) / YC TENDENCIA2
```

4.8.2 RATS routine: Female log(mortality) observed in Chile 2005, Japan 2006 and trend

```
* LOS CALCULOS CORRESPONDEN AL FILTRO DE HODRICK Y PRESCOTT Y SE
* APLICAN MEDIANTE LA IDEA DEL NUEVO ARTICULO Y EL FILTRO DE KALMAN CON SUAVIZAMIENTO.
**********************************************************************
* AQUI SE ESPECIFICA LA LONGITUD DE LA SERIE DE MORTALIDAD.
*
DECLARE REAL N1
COMPUTE INI = 1                           ;*DATO REQ.
COMPUTE FIN = 101                         ;*DATO REQ.
COMPUTE N = FIN-INI+1
COMPUTE N1= FIN-INI+1
DECLARE REAL SUAF%

OPEN DATA C:\3_ST\Finales\dataCHF.xls                 ;*ARCH. REQ.
DATA(FORMAT=xls,ORG=OBS) / CHF

OPEN DATA C:\3_ST\Finales\dataJPF.xls                 ;*ARCH. REQ.
DATA(FORMAT=xls,ORG=OBS) / JPF

*open styles thicklines.txt
*grparm(import=styles)

LABELS CHF JPF ;*Etiquetas para las variables DATOS y TENDENCIA
# 'CH FEMALE 2005' 'JPN FEMALE 2006'
GRAPH(Style=Line,$
key=lorigth,vlabel='Log(qx)',hlabel='Age x') 2
# CHF
# JPF

DECLARE RECTANGULAR IN K2
DIM IN(N,N)
COMPUTE IN=%identity(N)
COMPUTE K2=%zeros(N-2,N)
DO J=1,N
        DO I=1,N-2
   COMPUTE K2(I,I)=1
                COMPUTE K2(I,I+1)=-2
   COMPUTE K2(I,I+2)=1
  END DO
END DO

************************
```

```
********STEP 1***********
**ESTIMACION DE LAMBDA1**

DECLARE REAL LAMBDA1
DECLARE REAL NMIN
DECLARE REAL MAXS

COMPUTE SUAF% = 75 ;*suavidad final deseada por el usuario
COMPUTE NMIN = 2/(1-(SUAF%/100)) ;*número mínimo de observaciones para aspirar a la suavidad deseada
COMPUTE MAXS = 1-(2/N1)    ;*máxima suavidad que se puede alcanzar

DISPLAY '                '
DISPLAY '           STEP 1'
DISPLAY 'N MIN            ' NMIN
DISPLAY 'MAX SMOOTHNESS    ' MAXS

COMPUTE LAMBDA1 = 1
COMPUTE PARC = 1-(%trace(inv(%identity(N)+1*tr(K2)*K2))/N)
COMPUTE SUAF = SUAF%/100
DISPLAY 'INITIAL SMOOTHNESS' PARC
DISPLAY 'DESIRED SMOOTHNESS' SUAF

WHILE PARC<SUAF {
            COMPUTE LAMBDA1 = LAMBDA1+1
            COMPUTE PARC = 1-(%trace(inv(%identity(N)+LAMBDA1*tr(K2)*K2))/N)
}
END WHILE

DISPLAY 'LAMBDA1          ' LAMBDA1
DISPLAY 'FINAL SMOOTHNESS  ' PARC

****METODO A TRAVES DEL FILTRO DE KALMAN
* AQUI INICIA EL CALCULO DE LA TENDENCIA CON EL FILTRO DE KALMAN.

DECLARE RECT a c                      ;*Declaracion de las matrices A y c
DECLARE FRML[VECT] y                           ;*Declaracion del vector y a ser estimado
DECLARE SYMM sw v0                             ;*Declaracion de matrices simétricas
DECLARE VECT x0 sv                    ;*Declaracion de vectores de valores iniciales
COMPUTE a = ||2.0,-1.0|1.0,0.0||      ;*Matriz de transicion de la ecuacion de estado
COMPUTE c  = ||1.0|0.0||              ;*Matriz de la ecuacion de medida
FRML y  = ||CHF||                       ;*Asignacion de los CHF observados al vector y
COMPUTE sw = ||1.0|0.0,0.0||          ;*Asignacion de la matriz simétrica de covarianzas de las w´s. Es ||1.0, 0.0|0.0,0.0||
COMPUTE sv = ||LAMBDA1||                   ;*Asignacion de la matriz simétrica de covarianzas de las v´s. Valor de sigma eta del
paper, final pag.189.
COMPUTE x0 = ||CHF(INI),CHF(INI)||                        ;*Matriz de valores iniciales
COMPUTE v0 = ||4.0*LAMBDA1|4.0*LAMBDA1,4.0*LAMBDA1||    ;*Da la covarianza de x0
DLM(a=a,c=c,y=y,sv=sv,sw=sw,x0=x0,sx0=v0,TYPE=SMOOTH) INI FIN STATES  ;*TYPE=SMOOTH se refiere al Kalman
smoother
SET TENDENCIA1 = STATES(t) (1)                           ;*Asignacion de la variable tendencia a partir del comando DLM
SET CICLO = CHF-TENDENCIA1          ;*Estimacion de la variable ciclo
LABELS CHF TENDENCIA1               ;*Etiquetas para las variables DATOS y TENDENCIA
# 'CH FEMALE 2005' 'TREND'

* AQUI SE GRAFICAN LOS RESULTADOS Y SE GUARDAN EN UN ARCHIVO DE EXCEL.
*
DISPLAY(STORE=SUBTIT) 'N =' ### N ', Smoothness =' ##.# SUAF% $
                  '%  y  Lambda1 =' #####.# LAMBDA1
GRAPH(Style=Line,SUBHEADER=SUBTIT,$
key=lorigth,vlabel='Log(qx)',hlabel='Age x') 2
# CHF
# TENDENCIA1
```

```
*
OPEN COPY C:\3_ST\Finales\CHF_S.XLS                    ;*ARCH. REQ.
COPY(FORMAT=XLS,ORG=OBS,DATES) / CHF TENDENCIA1

************************
********STEP 2***********

DECLARE REAL LAMBDA
DECLARE REAL ALFA
DECLARE REAL SUAF2%
DECLARE REAL NMIN2
DECLARE REAL MAXS2

COMPUTE SUAF2% = 70 ;*suavidad final deseada por el usuario
COMPUTE NMIN2 = 2/(1-(SUAF2%/100)) ;*número mínimo de observaciones para aspirar a la suavidad deseada
COMPUTE MAXS2 = 1-(2/N1)    ;*máxima suavidad que se puede alcanzar

DISPLAY '                '
DISPLAY '          STEP 2'
DISPLAY 'N MIN            ' NMIN2
DISPLAY 'MAX SMOOTHNESS    ' MAXS2

COMPUTE LAMBDA = 1
COMPUTE PARC2 = 1-(%trace(inv(%identity(N)+1*tr(K2)*K2))/N)
COMPUTE SUAF2 = SUAF2%/100

DISPLAY 'INITIAL SMOOTHNESS' PARC2
DISPLAY 'DESIRED SMOOTHNESS' SUAF2

WHILE PARC2<SUAF2 {
            COMPUTE LAMBDA = LAMBDA+1
            COMPUTE PARC2 = 1-(%trace(inv(%identity(N)+LAMBDA*tr(K2)*K2))/N)
}
END WHILE

DISPLAY 'LAMBDA           ' LAMBDA
DISPLAY 'FINAL SMOOTHNESS  ' PARC2

COMPUTE ALFAU = 0.5
COMPUTE OPC = 1              ; *WITH OPC = 1, ALFA IS ESTIMATED
                                                                                    *WITH OPC = 2,
ALFA IS SELECTED BY THE USER
COMPUTE ALFA = %IF(OPC==1,LAMBDA/LAMBDA1,ALFAU)
COMPUTE SUAF2% = %IF(OPC==1,SUAF2%,(1-(%trace(inv(%identity(N)+LAMBDA1*ALFAU*tr(K2)*K2))/N))*100)

DISPLAY 'SELECTED OPTION           ' OPC
DISPLAY 'ALFA                                ' ALFA
DISPLAY 'SUAF                                ' SUAF2

COMPUTE LAMBDA2 =(LAMBDA1/LAMBDA)-1

DISPLAY 'LAMBDA2                             ' LAMBDA2

DECLARE VECTOR YC1(N)

DO i=1,N
COMPUTE YC1(i) = ALFA*%IF(i<=110,CHF(i),TENDENCIA1(i))+(1-ALFA)*JPF(i)
END DO i
SET YC 1 N =YC1(t)
PRINT 1 N-6 YC
```

```
* AQUI INICIA EL CALCULO DE LA TENDENCIA CON EL FILTRO DE KALMAN.

DECLARE RECT a c                          ;*Declaracion de las matrices A y c
DECLARE FRML[VECT] y                      ;*Declaracion del vector y a ser estimado
DECLARE SYMM sw v0                        ;*Declaracion de matrices simétricas
DECLARE VECT x0 sv                        ;*Declaracion de vectores de valores iniciales
COMPUTE a = ||2.0,-1.0|1.0,0.0||          ;*Matriz de transicion de la ecuacion de estado
COMPUTE c  = ||1.0|0.0||                  ;*Matriz de la ecuacion de medida
FRML y  = ||YC||                          ;*Asignacion de los YC observados al vector y
COMPUTE sw = ||1.0|0.0,0.0||              ;*Asignacion de la matriz simétrica de covarianzas de las w´s. Es ||1.0, 0.0|0.0,0.0||
COMPUTE sv = ||LAMBDA||                   ;*Asignacion de la matriz simétrica de covarianzas de las v´s. Valor de sigma eta del
paper, final pag.189.
COMPUTE x0 = ||YC(INI),YC(INI)||          ;*Matriz de valores iniciales
COMPUTE v0 = ||4.0*LAMBDA|4.0*LAMBDA,4.0*LAMBDA||    ;*Da la covarianza de x0
DLM(a=a,c=c,y=y,sv=sv,sw=sw,x0=x0,sx0=v0,TYPE=SMOOTH) INI FIN STATES  ;*TYPE=SMOOTH se refiere al Kalman
smoother
SET TENDENCIA2 = STATES(t) (1)            ;*Asignacion de la variable tendencia a partir del comando DLM
SET CICLO = YC-TENDENCIA2                 ;*Estimacion de la variable ciclo a partir del comando DLM

LABELS CHF TENDENCIA2 JPF                 ;*Etiquetas para las variables YC y TENDENCIA
# 'CH female 2005' 'Trend' 'JPN female 2006'
DISPLAY(STORE=SUBTIT) 'N =' ### N ', Smoothness =' ##.# SUAF2% $
                  '%, Lambda =' #####.# LAMBDA ', Alfa =' #.## ALFA
GRAPH(Style=Line,key=lorigth,$
vlabel='Log(qx)',hlabel='Age x',SUBHEADER=SUBTIT) 3
# CHF
# TENDENCIA2
# JPF
*
OPEN COPY C:\3_ST\Finales\YC_S.XLS                  ;*ARCH. REQ.
COPY(FORMAT=XLS,ORG=OBS,DATES) / YC TENDENCIA2
```

4.8.3 RATS routine: Log(mortality) observed by period 2000 and cohort 2010 for the US and trend

```
* LOS CALCULOS CORRESPONDEN AL FILTRO DE HODRICK Y PRESCOTT Y SE
* APLICAN MEDIANTE LA IDEA DEL NUEVO ARTICULO Y EL FILTRO DE KALMAN CON SUAVIZAMIENTO.
*
*******************************************************************
*
* AQUI SE ESPECIFICA LA LONGITUD DE LA SERIE DE MORTALIDAD.
*
DECLARE REAL N1
COMPUTE INI = 1                           ;*DATO REQ.
COMPUTE FIN = 120                         ;*DATO REQ.
COMPUTE N = FIN-INI+1
COMPUTE N1= FIN-INI+1
DECLARE REAL SUAF%

OPEN DATA C:\3_ST\Finales\dataUSAM.xls              ;*ARCH. REQ.
DATA(FORMAT=xls,ORG=OBS) / USAPM USACM

*open styles thicklines.txt
*grparm(import=styles)

LABELS USAPM USACM                        ;*Etiquetas para las variables DATOS y TENDENCIA
# 'PERIOD 2000' 'COHORT 2010'
GRAPH(Style=Line,$
key=lorigth,vlabel='Log(qx)',hlabel='Age x') 2
# USAPM
```

```
# USACM

DECLARE RECTANGULAR IN K2
DIM IN(N,N)
COMPUTE IN=%identity(N)
COMPUTE K2=%zeros(N-2,N)
DO J=1,N
          DO I=1,N-2
    COMPUTE K2(I,I)=1
                    COMPUTE K2(I,I+1)=-2
    COMPUTE K2(I,I+2)=1
  END DO
END DO

*************************
********STEP 1***********
**ESTIMACION DE LAMBDA1**

DECLARE REAL LAMBDA1
DECLARE REAL NMIN
DECLARE REAL MAXS

COMPUTE SUAF% = 80 ;*suavidad final deseada por el usuario
COMPUTE NMIN = 2/(1-(SUAF%/100)) ;*número mínimo de observaciones para aspirar a la suavidad deseada
COMPUTE MAXS = 1-(2/N1)    ;*máxima suavidad que se puede alcanzar

DISPLAY '                '
DISPLAY '           STEP 1'
DISPLAY 'N MIN             ' NMIN
DISPLAY 'MAX SMOOTHNESS    ' MAXS

COMPUTE LAMBDA1 = 1
COMPUTE PARC = 1-(%trace(inv(%identity(N)+1*tr(K2)*K2))/N)
COMPUTE SUAF = SUAF%/100
DISPLAY 'INITIAL SMOOTHNESS' PARC
DISPLAY 'DESIRED SMOOTHNESS' SUAF

WHILE PARC<SUAF {
          COMPUTE LAMBDA1 = LAMBDA1+1
          COMPUTE PARC = 1-(%trace(inv(%identity(N)+LAMBDA1*tr(K2)*K2))/N)
}
END WHILE

DISPLAY 'LAMBDA1          ' LAMBDA1
DISPLAY 'FINAL SMOOTHNESS  ' PARC

****METODO A TRAVES DEL FILTRO DE KALMAN
* AQUI INICIA EL CALCULO DE LA TENDENCIA CON EL FILTRO DE KALMAN.

DECLARE RECT a c                    ;*Declaracion de las matrices A y c
DECLARE FRML[VECT] y                          ;*Declaracion del vector y a ser estimado
DECLARE SYMM sw v0                            ;*Declaracion de matrices simétricas
DECLARE VECT x0 sv                  ;*Declaracion de vectores de valores iniciales
COMPUTE a = ||2.0,-1.0|1.0,0.0||    ;*Matriz de transicion de la ecuacion de estado
COMPUTE c  = ||1.0|0.0||            ;*Matriz de la ecuacion de medida
FRML y  = ||USAPM||                   ;*Asignacion de los USAPM observados al vector y
COMPUTE sw = ||1.0|0.0,0.0||        ;*Asignacion de la matriz simétrica de covarianzas de las w´s. Es ||1.0, 0.0|0.0,0.0||
COMPUTE sv = ||LAMBDA1||               ;*Asignacion de la matriz simétrica de covarianzas de las v´s. Valor de sigma eta del
paper, final pag.189.
COMPUTE x0 = ||USAPM(INI),USAPM(INI)||              ;*Matriz de valores iniciales
COMPUTE v0 = ||4.0*LAMBDA1|4.0*LAMBDA1,4.0*LAMBDA1||    ;*Da la covarianza de x0
```

```
DLM(a=a,c=c,y=y,sv=sv,sw=sw,x0=x0,sx0=v0,TYPE=SMOOTH) INI FIN STATES  ;*TYPE=SMOOTH se refiere al Kalman
smoother
SET TENDENCIA1 = STATES(t) (1)                    ;*Asignacion de la variable tendencia a partir del comando DLM
SET CICLO = USAPM-TENDENCIA1          ;*Estimacion de la variable ciclo
LABELS USAPM TENDENCIA1               ;*Etiquetas para las variables DATOS y TENDENCIA
# 'USA MAL PER 2000' 'TREND'
*
* AQUI SE GRAFICAN LOS RESULTADOS Y SE GUARDAN EN UN ARCHIVO DE EXCEL.
*
DISPLAY(STORE=SUBTIT) 'N =' ### N ', Smoothness =' ##.# SUAF% $
               '%  y  Lambda1 =' #####.# LAMBDA1
GRAPH(Style=Line,SUBHEADER=SUBTIT,$
key=lorigth,vlabel='Log(qx)',hlabel='Age x') 2
# USAPM
# TENDENCIA1
*
OPEN COPY C:\3_ST\Finales\USAPM_S.XLS                  ;*ARCH. REQ.
COPY(FORMAT=XLS,ORG=OBS,DATES) / USAPM TENDENCIA1

*************************
********STEP 2***********

DECLARE REAL LAMBDA
DECLARE REAL ALFA
DECLARE REAL SUAF2%
DECLARE REAL NMIN2
DECLARE REAL MAXS2

COMPUTE SUAF2% = 75 ;*suavidad final deseada por el usuario
COMPUTE NMIN2 = 2/(1-(SUAF2%/100)) ;*número mínimo de observaciones para aspirar a la suavidad deseada
COMPUTE MAXS2 = 1-(2/N1)    ;*máxima suavidad que se puede alcanzar

DISPLAY '                '
DISPLAY '          STEP 2'
DISPLAY 'N MIN           ' NMIN2
DISPLAY 'MAX SMOOTHNESS    ' MAXS2

COMPUTE LAMBDA = 1
COMPUTE PARC2 = 1-(%trace(inv(%identity(N)+1*tr(K2)*K2))/N)
COMPUTE SUAF2 = SUAF2%/100

DISPLAY 'INITIAL SMOOTHNESS' PARC2
DISPLAY 'DESIRED SMOOTHNESS' SUAF2

WHILE PARC2<SUAF2 {
            COMPUTE LAMBDA = LAMBDA+1
            COMPUTE PARC2 = 1-(%trace(inv(%identity(N)+LAMBDA*tr(K2)*K2))/N)
}
END WHILE

DISPLAY 'LAMBDA          ' LAMBDA
DISPLAY 'FINAL SMOOTHNESS  ' PARC2

COMPUTE ALFAU = 0.6
COMPUTE OPC = 2             ; *WITH OPC = 1, ALFA IS ESTIMATED
                                                                                        *WITH OPC = 2,
ALFA IS SELECTED BY THE USER
COMPUTE ALFA = %IF(OPC==1,LAMBDA/LAMBDA1,ALFAU)
COMPUTE SUAF2% = %IF(OPC==1,SUAF2%,(1-(%trace(inv(%identity(N)+LAMBDA1*ALFAU*tr(K2)*K2))/N))*100)

DISPLAY 'SELECTED OPTION           ' OPC
```

```
DISPLAY 'ALFA                                   ' ALFA
DISPLAY 'SUAF                                   ' SUAF2

COMPUTE LAMBDA2 =(LAMBDA1/LAMBDA)-1

DISPLAY 'LAMBDA2                                ' LAMBDA2

DECLARE VECTOR YC1(N)

DO i=1,N
COMPUTE YC1(i) = ALFA*%IF(i<=110,USAPM(i),TENDENCIA1(i))+(1-ALFA)*USACM(i)
END DO i
SET YC 1 N =YC1(t)
PRINT 1 N YC

* AQUI INICIA EL CALCULO DE LA TENDENCIA CON EL FILTRO DE KALMAN.

DECLARE RECT a c                     ;*Declaracion de las matrices A y c
DECLARE FRML[VECT] y                          ;*Declaracion del vector y a ser estimado
DECLARE SYMM sw v0                            ;*Declaracion de matrices simétricas
DECLARE VECT x0 sv                   ;*Declaracion de vectores de valores iniciales
COMPUTE a = ||2.0,-1.0|1.0,0.0||     ;*Matriz de transicion de la ecuacion de estado
COMPUTE c  = ||1.0|0.0||             ;*Matriz de la ecuacion de medida
FRML y  = ||YC||                       ;*Asignacion de los YC observados al vector y
COMPUTE sw = ||1.0|0.0,0.0||         ;*Asignacion de la matriz simétrica de covarianzas de las w´s. Es ||1.0, 0.0|0.0,0.0||
COMPUTE sv = ||LAMBDA||                 ;*Asignacion de la matriz simétrica de covarianzas de las v´s. Valor de sigma eta del
paper, final pag.189.
COMPUTE x0 = ||YC(INI),YC(INI)||                ;*Matriz de valores iniciales
COMPUTE v0 = ||4.0*LAMBDA|4.0*LAMBDA,4.0*LAMBDA||    ;*Da la covarianza de x0
DLM(a=a,c=c,y=y,sv=sv,sw=sw,x0=x0,sx0=v0,TYPE=SMOOTH) INI FIN STATES  ;*TYPE=SMOOTH se refiere al Kalman
smoother
SET TENDENCIA2 = STATES(t) (1)                        ;*Asignacion de la variable tendencia a partir del comando DLM
SET CICLO = YC-TENDENCIA2           ;*Estimacion de la variable ciclo a partir del comando DLM

LABELS USAPM TENDENCIA2 USACM           ;*Etiquetas para las variables YC y TENDENCIA
# 'US male per 2000' 'Trend' 'US male coh 2010'
DISPLAY(STORE=SUBTIT) 'N =' ### N ', Smoothness =' ##.# SUAF2% $
                  '%, Lambda =' #####.# LAMBDA ', ALPHA =' #.## ALFA
GRAPH(Style=Line,key=lorigth,$
vlabel='Log(qx)',hlabel='Age x',SUBHEADER=SUBTIT) 3
# USAPM
# TENDENCIA2
# USACM
*
OPEN COPY C:\3_ST\Finales\YC_S.XLS                        ;*ARCH. REQ.
COPY(FORMAT=XLS,ORG=OBS,DATES) / YC TENDENCIA2
```

4.8.4 RATS routine: Log(mortality) observed in the XIX Century Mexico City and trend

```
* LOS CALCULOS CORRESPONDEN AL FILTRO DE HODRICK Y PRESCOTT Y SE
* APLICAN MEDIANTE LA IDEA DEL NUEVO ARTICULO Y EL FILTRO DE KALMAN CON SUAVIZAMIENTO.
*
*******************************************************************
*
* AQUI SE ESPECIFICA LA LONGITUD DE LA SERIE DE MORTALIDAD.
*
DECLARE REAL N1
COMPUTE INI = 1                          ;*DATO REQ.
COMPUTE FIN = 19                        ;*DATO REQ.
COMPUTE N = FIN-INI+1
COMPUTE N1= FIN-INI+1
```

```
DECLARE REAL SUAF%

OPEN DATA C:\3_ST\Finales\MEXPP.xls                    ;*ARCH. REQ.
DATA(FORMAT=xls,ORG=OBS) / PAN PAR

*open styles thicklines.txt
*grparm(import=styles)

grparm(font="Symbol") axislabels 16

COMPUTE [VECT[STRINGS]] FLAB=$
||"0-1","","5-9","","15-19","","25-29","","35-39","",$
"45-49","","55-59","","65-69","","75-79","","85+"||

LABELS PAN PAR ;*Etiquetas para las variables DATOS y TENDENCIA
# 'Paula cemetery' 'Sta Mary parish'
GRAPH(Style=Line,$
key=lorigth,vlabel='Log(qx)',hlabel='Age groups', $
xlabel=FLAB) 2
# PAN
# PAR

DECLARE RECTANGULAR IN K2
DIM IN(N,N)
COMPUTE IN=%identity(N)
COMPUTE K2=%zeros(N-2,N)
DO J=1,N
          DO I=1,N-2
    COMPUTE K2(I,I)=1
                    COMPUTE K2(I,I+1)=-2
    COMPUTE K2(I,I+2)=1
  END DO
END DO

*************************
********STEP 1***********
**ESTIMACION DE LAMBDA1**

DECLARE REAL LAMBDA1
DECLARE REAL NMIN
DECLARE REAL MAXS

COMPUTE SUAF% = 80 ;*suavidad final deseada por el usuario
COMPUTE NMIN = 2/(1-(SUAF%/100)) ;*número mínimo de observaciones para aspirar a la suavidad deseada
COMPUTE MAXS = 1-(2/N1)    ;*máxima suavidad que se puede alcanzar

DISPLAY '                '
DISPLAY '          STEP 1'
DISPLAY 'N MIN            ' NMIN
DISPLAY 'MAX SMOOTHNESS    ' MAXS

COMPUTE LAMBDA1 = 1
COMPUTE PARC = 1-(%trace(inv(%identity(N)+1*tr(K2)*K2))/N)
COMPUTE SUAF = SUAF%/100
DISPLAY 'INITIAL SMOOTHNESS' PARC
DISPLAY 'DESIRED SMOOTHNESS' SUAF

WHILE PARC<SUAF {
          COMPUTE LAMBDA1 = LAMBDA1+1
          COMPUTE PARC = 1-(%trace(inv(%identity(N)+LAMBDA1*tr(K2)*K2))/N)
}
```

```
END WHILE

DISPLAY 'LAMBDA1          ' LAMBDA1
DISPLAY 'FINAL SMOOTHNESS  ' PARC

****METODO A TRAVES DEL FILTRO DE KALMAN
* AQUI INICIA EL CALCULO DE LA TENDENCIA CON EL FILTRO DE KALMAN.

DECLARE RECT a c                    ;*Declaracion de las matrices A y c
DECLARE FRML[VECT] y                ;*Declaracion del vector y a ser estimado
DECLARE SYMM sw v0                  ;*Declaracion de matrices simétricas
DECLARE VECT x0 sv                  ;*Declaracion de vectores de valores iniciales
COMPUTE a = ||2.0,-1.0|1.0,0.0||    ;*Matriz de transicion de la ecuacion de estado
COMPUTE c  = ||1.0|0.0||            ;*Matriz de la ecuacion de medida
FRML y  = ||PAN||                   ;*Asignacion de los PAN observados al vector y
COMPUTE sw = ||1.0|0.0,0.0||        ;*Asignacion de la matriz simétrica de covarianzas de las w´s. Es ||1.0, 0.0|0.0,0.0||
COMPUTE sv = ||LAMBDA1||            ;*Asignacion de la matriz simétrica de covarianzas de las v´s. Valor de sigma eta del
paper, final pag.189.
COMPUTE x0 = ||PAN(INI),PAN(INI)||                  ;*Matriz de valores iniciales
COMPUTE v0 = ||4.0*LAMBDA1|4.0*LAMBDA1,4.0*LAMBDA1||    ;*Da la covarianza de x0
DLM(a=a,c=c,y=y,sv=sv,sw=sw,x0=x0,sx0=v0,TYPE=SMOOTH) INI FIN STATES  ;*TYPE=SMOOTH se refiere al Kalman
smoother
SET TENDENCIA1 = STATES(t) (1)                      ;*Asignacion de la variable tendencia a partir del comando DLM
SET CICLO = PAN-TENDENCIA1          ;*Estimacion de la variable ciclo
LABELS PAN TENDENCIA1               ;*Etiquetas para las variables DATOS y TENDENCIA
# 'STA PAULA CEMETERY' 'TREND'
*
* AQUI SE GRAFICAN LOS RESULTADOS Y SE GUARDAN EN UN ARCHIVO DE EXCEL.
*
DISPLAY(STORE=SUBTIT) 'N =' ### N ', Smoothness =' ##.# SUAF% $
                  '%  y  Lambda1 =' #####.# LAMBDA1
GRAPH(Style=Line,SUBHEADER=SUBTIT,$
key=lorigth,vlabel='Log(qx)',hlabel='Age groups', $
xlabel=FLAB) 2
# PAN
# TENDENCIA1
*
OPEN COPY C:\3_ST\Finales\PAN_S.XLS                 ;*ARCH. REQ.
COPY(FORMAT=XLS,ORG=OBS,DATES) / PAN TENDENCIA1

*******STEP 2***********

DECLARE REAL LAMBDA
DECLARE REAL ALFA
DECLARE REAL SUAF2%
DECLARE REAL NMIN2
DECLARE REAL MAXS2

COMPUTE SUAF2% = 75 ;*suavidad final deseada por el usuario
COMPUTE NMIN2 = 2/(1-(SUAF2%/100)) ;*número mínimo de observaciones para aspirar a la suavidad deseada
COMPUTE MAXS2 = 1-(2/N1)    ;*máxima suavidad que se puede alcanzar

DISPLAY '                '
DISPLAY '          STEP 2'
DISPLAY 'N MIN           ' NMIN2
DISPLAY 'MAX SMOOTHNESS    ' MAXS2

COMPUTE LAMBDA = 1
COMPUTE PARC2 = 1-(%trace(inv(%identity(N)+1*tr(K2)*K2))/N)
COMPUTE SUAF2 = SUAF2%/100
```

```
DISPLAY 'INITIAL SMOOTHNESS' PARC2
DISPLAY 'DESIRED SMOOTHNESS' SUAF2

WHILE PARC2<SUAF2 {
        COMPUTE LAMBDA = LAMBDA+1
        COMPUTE PARC2 = 1-(%trace(inv(%identity(N)+LAMBDA*tr(K2)*K2))/N)
}
END WHILE

DISPLAY 'LAMBDA          ' LAMBDA
DISPLAY 'FINAL SMOOTHNESS  ' PARC2

COMPUTE ALFAU = 0.8
COMPUTE OPC = 2              ; *WITH OPC = 1, ALFA IS ESTIMATED
                                                                *WITH OPC = 2,
ALFA IS SELECTED BY THE USER
COMPUTE ALFA = %IF(OPC==1,LAMBDA/LAMBDA1,ALFAU)
COMPUTE SUAF2% = %IF(OPC==1,SUAF2%,(1-(%trace(inv(%identity(N)+LAMBDA1*ALFAU*tr(K2)*K2))/N))*100)

DISPLAY 'SELECTED OPTION          ' OPC
DISPLAY 'ALFA                     ' ALFA
DISPLAY 'SUAF                     ' SUAF2

COMPUTE LAMBDA2 =(LAMBDA1/LAMBDA)-1
*RECALCULA LAMBDA
DISPLAY 'LAMBDA2                  ' LAMBDA2

DECLARE VECTOR YC1(N)

DO i=1,N
COMPUTE YC1(i) = ALFA*%IF(i<=110,PAN(i),TENDENCIA1(i))+(1-ALFA)*PAR(i)
END DO i
SET YC 1 N =YC1(t)
PRINT 1 N-6 YC

* AQUI INICIA EL CALCULO DE LA TENDENCIA CON EL FILTRO DE KALMAN.

DECLARE RECT a c                  ;*Declaracion de las matrices A y c
DECLARE FRML[VECT] y              ;*Declaracion del vector y a ser estimado
DECLARE SYMM sw v0                ;*Declaracion de matrices simétricas
DECLARE VECT x0 sv                ;*Declaracion de vectores de valores iniciales
COMPUTE a = ||2.0,-1.0|1.0,0.0||  ;*Matriz de transicion de la ecuacion de estado
COMPUTE c  = ||1.0|0.0||          ;*Matriz de la ecuacion de medida
FRML y  = ||YC||                  ;*Asignacion de los YC observados al vector y
COMPUTE sw = ||1.0|0.0,0.0||      ;*Asignacion de la matriz simétrica de covarianzas de las w´s. Es ||1.0, 0.0|0.0,0.0||
COMPUTE sv = ||LAMBDA||           ;*Asignacion de la matriz simétrica de covarianzas de las v´s. Valor de sigma eta del
paper, final pag.189.
COMPUTE x0 = ||YC(INI),YC(INI)||              ;*Matriz de valores iniciales
COMPUTE v0 = ||4.0*LAMBDA|4.0*LAMBDA,4.0*LAMBDA||    ;*Da la covarianza de x0
DLM(a=a,c=c,y=y,sv=sv,sw=sw,x0=x0,sx0=v0,TYPE=SMOOTH) INI FIN STATES  ;*TYPE=SMOOTH se refiere al Kalman
smoother
SET TENDENCIA2 = STATES(t) (1)             ;*Asignacion de la variable tendencia a partir del comando DLM
SET CICLO = YC-TENDENCIA2          ;*Estimacion de la variable ciclo a partir del comando DLM

LABELS PAN TENDENCIA2 PAR          ;*Etiquetas para las variables YC y TENDENCIA
# 'Sta Paula cemetery' 'Trend' 'Sta Maria parish'
DISPLAY(STORE=SUBTIT) 'N =' ### N ', Smoothness =' ##.# SUAF2% $
        '%, Lambda =' #####.# LAMBDA ', Alfa =' #.## ALFA
GRAPH(Style=Line,key=lorigth,$
vlabel='Log(qx)',hlabel='Age groups',SUBHEADER=SUBTIT, $
xlabel=FLAB) 3
```

```
# PAN
# TENDENCIA2
# PAR
*
OPEN COPY C:\3_ST\Finales\YC_S.XLS                    ;*ARCH. REQ.
COPY(FORMAT=XLS,ORG=OBS,DATES) / YC TENDENCIA2
```

Chapter 5. Conclusions and further research

This thesis exploits the opportunity of using or developing specialized statistical methodologies with the main purpose of solving problems dealing with univariate or multivariate time series in the demographic or the actuarial field. The emphasis is placed on how both the univariate and the multivariate statistical approaches exposed in the chapters, can be useful and adequate for making decisions in a demographic context. It is evident the possibility of using different statistical or econometric softwares such as: E-Views, Matlab, R, RATS, whose main application is not necessarily for demographic analysis. Some general conclusions generated from the chapters are now established and finally, some lines for future research are suggested.

In the Chapter 2, once the temporal disaggregation technique chosen is applied to the demographic time series, it was found that the most time-consuming activity is the generation of an appropriate preliminary series. This task is much simpler to perform in other contexts, as in economics, due to the availability of economic indicators. The multiple restricted forecasting was applied once a set of compatibility tests were conducted in order to know the feasibility of the proposed targets in a program of official population goals for Mexico (the targets were evaluated on the basis of population growth). From the statistical results so derived, another set of targets were suggested, in concordance with the behavior observed in the population series. This analysis leads to the suggestion that demographic goals should be suggested in a more objective way and preferably based on their empirical feasibility. Finally, the approaches presented here can be used primarily in developing countries or in other geographical units where similar problems to the ones presented here appear. It was concluded that the programs of population growth could be established with statistical support if they apply a procedure similar to the one employed here to obtain empirical evidence.

The method proposed in the Chapter 3 to estimate trends in mortality series arises from the idea that the user can fix a desired percentage of smoothness for the trend beforehand. With such a proposal it is also possible to estimate missing data or produce forecasts. The method as applied to mortality rates can be useful for diagnosis and decision making in the insurance industry or in the context of establishing population policies. The possibility of comparing trends of smoothed mortality rates starts with the calculation of a smoothness index whose properties are exposed here. With the aid of this index it is feasible to identify, among other things, the relationship between one dimensional smoothness and two-dimensional smoothness. Besides, the theoretical results have a solid mathematical and statistical support. In particular, the equivalence between the approaches already known in the literature and that obtained with Generalized Least Squares is illustrated. The calculations can be performed efficiently without having to invest in higher-dimensional matrices using a rule already established by empirical workers. Some examples that illustrate the results that can be obtained in practice with the proposed methodology are presented.

In the Chapter 4, the proposed methodology is useful for estimating trends in mortality when considering fidelity to data, smoothness of the estimates and information on structure from a given mortality table. It allows the analyst to control both the percentage of smoothness and the structure according to his/her interests, in order to achieve comparability. One of the most noticeable advantage of the methodology proposed here is the possibility that the analyst can give more credibility to a source of information than to the other. Some circumstances that might arise when implementing this proposal are the presence of missing data or data bases with different sizes. Both of those situations are overcome by using the Kalman filter (KF). Finally, it is noted that the application of the proposed methodology can be performed on other types of demographic indicators, such as fertility, marriage, divorce and migration.

From the developments in this thesis, several lines of future research are displayed. One is the analysis of populations by cohort. That is, if the partial series of population cohorts and the number of total population are available, it would be useful to include the series of cohorts and use either one of the following approaches: a) consider that the sum of the predictions of the cohorts is consistent with the predictions of the total, or b) assume that there is a *factor* in all cohorts, leading to a factor model building to generate predictions by combining data from each series and the total (this leads to a dynamic factor analysis). It would also be convenient to examine the relationship between these two approaches. Moreover, this topic could be related with the idea of smoothness, where the analyst decides a desired level of smoothness to promote comparability with other trends in mortality or other demographic indicators. This idea would suggest other smoothness indices and the corresponding study of their properties.

On the other hand, it would be interesting to explore and analyze, in terms of restricted forecasts for the univariate or multivariate cases, demographic events under the presence of the so-called volatility, as in the family of Auto-Regressive Conditional Heteroskedasticity (ARCH) models. Among the phenomena producing conditional heteroskedasticity, the following could be identified: special types of migrations; specific causes of mortality, as car accidents; morbidity through the spread of epidemics; the level of economically active population in different geographical areas through periodic surveys; and so on.

Another line of research is the combination of information regarding theoretical laws of mortality (parametric models) and general structures of mortality, with data coming from developed countries or developing regions, where the analyst may assign certain level of credibility to a particular source. This situation could also be generalized to the presence of more than two sources of information. For this purpose, it might be appropriate to use nonlinear optimization, to define appropriate loss functions and, very probably, to develop skills in using computer programs necessary for performing the required calculations.

Regarding the proposal of the third chapter, the methodology could also be generalized to the two-dimensional case which, as it happened with the one-dimensional case, could require the derivation of some new theoretical results to relate the different smoothing parameters. Moreover, it would be appropriate to apply the technique to generate estimates of mortality surfaces, restricted to the experience and beliefs that the analyst considers appropriate, in order to graduate information and to support comparability. In practical terms, it could be necessary to make some proposal where the methodology be applied by pieces of the series of mortality, within the age range, for both the one-dimensional and the two-dimensional cases. This need can occur when the analyst wants an estimate that gets closer to the demographic structure in certain range and keeps the rest in a balanced manner between the different data sources.

Bibliography

Alonso, A. M., Peña, D. and Rodríguez, J. (2009) A Methodology for Population Projections: An Application to Spain, Preprint submitted to CSDA.

Alonso, A. M. (2008) *Predicción de tablas de mortalidad dinámicas mediante un procedimiento bootstrap*, Editorial MAPFRE, Madrid. ISBN 978-84-9844-101-7.

Bell, W. (1997) Comparing and assessing time series methods for forecasting age-specific fertility and mortality rates. *Journal of Official Statistics*, 13: 279-303.

Boone, L. (2000) Comparing semi-structural methods to estimate unobserved variables: the HPMV and Kalman Filters approaches, *Economics Department Working Papers*, No. 240, OCDE.

Bourgeois-Pichat, J. (1952) Essai sur la mortalité 'biologique' de l'homme, *Population*, 7, 3: 381-394.

Brouhns, N., Denuit, M. and Vermunt, J. (2002) A Poisson log-bilinear regression approach to the construction of projected life tables, *Insurance: Mathematics and Economics*, 31: 373-93.

Brücker, H., Siliverstovs, B. and Trübswetter, P. (2003) International Migration to Germany: Estimation of a Time-Series Model and Inference in Panel Cointegration, Discussion Papers of DIW Berlin 391, DIW Berlin, German Institute for Economic Research.

Carter, L. and Lee. R. (1986) Joint forecasts of US marital fertility, nuptiality, births and marriages using time series models, *Journal of the American Statistical Association*.

Chow, G. and Lin, A. (1971) Best linear unbiased interpolation, distribution and extrapolation of time series by related series, *The Review of Economics and Statistics,* 53: 372-375.

Clayton, D. and Schifflers, E. (1987) Models for temporal variation in cancer rates. II: Age-period-cohort models, *Statistics in Medicine*, 6: 469-81.

Clements, M. and Galvão, A. (2008) Macroeconomic Forecasting With Mixed-Frequency Data: Forecasting Output Growth in the United States, *Journal of Business & Economic Statistics*, 26: 546-554.

Cleveland, W. and Devlin, S. (1988) Locally weighted regression: an approach to regression analysis by local fitting, *Journal of the American Statistical Association*, 83: 597-610.

Cornwell, D. (2009) An Exploratory Time Series Analysis of Apprehensions and Linewatch Hours on the Southwest Border, Working paper, Office of Immigration Statistics.

Currie, I. and Durban, M. (2002) Flexible smoothing with P-splines: a unified approach, *Statistical Modelling*, 2: 333-349.

Currie, I., Durban, M. and Eilers, P. (2004) Smoothing and forecasting mortality rates, *Statistical Modelling*, 4: 279-298.

De Beer, J. (1985) A time-series model for cohort data, *Journal of the American Statistical Association*, 80: 525-530.

De Beer, J. (1989) Projecting age-specific fertility by using time series models, *European Journal of Population*, 5: 315-346.

De Beer, J. (1992) General time-series models for forecasting fertility, 148-175 in N. Keilman and H. Cruijsen (eds.) National Population Forecasting in Industrialized Countries. Amsterdam etc.: Swets & Zeitlinger.

De Boor, C. (2001) *A practical guide to splines*. New York: Springer.

Dierckx, P. (1993) *Curve and surface fitting with splines*. Oxford: Clarendon Press.

Di Fonzo, T. (1994) Temporal disaggregation of a system of time series when the aggregate is known: optimal vs. adjustment methods, Workshop on Quarterly National Accounts, Paris, 1-22.

Di Fonzo, T. (1990) The estimation of m disaggregate time series when contemporaneous and temporal aggregates are known, *The Review of Economics and Statistics,* 72: 178-182.

Dufour, J. (1985) Unbiasedness of predictions from estimated vector autoregressions, *Econometric Theory*, 1: 387– 402.

Durban, M., Currie, I. and Kempton, R. (2001) Adjusting for fertility and competition in variety trials (postscript), Working paper.

Eilers, P. and Marx, B. (1996) Flexible smoothing with B-splines and penalties, *Statistical Science*, 11, 89-121.

George, V., Smith, S., Swason, D. and Tayman, J.(2004) "The Methods and Materials of Demography" in chapter 21, *Population Projections*, edited by J. Siegel and Swanson, D., Elsevier Academic Press.

Ghysels, E., Santa-Clara, P. and Valkanov, R. (2006) Predicting volatility: Getting the most out of return data sampled at different frequencies, *Journal of Econometrics*, 131: 59-95.

Girosi, F. and King, G. (2004) Demographic forecasting, unpublished book manuscript. Downloadable at http://gking. harvard. edu/files/smooth. pdf.

Gobierno del DF y Consejo de Población del Distrito Federal (1997) Programas de Población del Distrito Federal 2001-2006: Nivel normativo (Metas). Dirección de Política Poblacional. Available at http://www.df.gob.mx/secretarias/social/copodf/index.html/[30 June, 2006].

Goldsteina, E., Dushoffb, J., Mad, J., Plotkine, J., Earnc, D. and Lipsitcha, M. (2009) Reconstructing influenza incidence by deconvolution of daily mortality time series, Edited by Burton H. Singer, Princeton University, Princeton, NJ.

Gómez, V. (1999) Three equivalent methods for filtering finite nonstationary time series, *Journal of Business Economics and Statistics*, 17: 109–166.

Gómez, N. and Guerrero, V. M. (2006) Restricted forecasting with VAR models: an analysis of a test for joint compatibility between restrictions and forecasts, *International Journal of Forecasting,* 22: 751– 770.

Gómez, V., Maravall, A. and Peña, D. (1999) Missing observations in ARIMA models: skipping approach versus additive outlier approach*, Journal of Econometrics*, 88: 341-363.

Guerrero, V. M. (2007) Time series smoothing by penalized least squares, *Statistics and Probability Letters*, 77: 225–1234.

Guerrero, V. M. (2008) Estimating Trends with Percentage of Smoothness Chosen by the User, *International Statistical Review*, 76, 2: 187–202.

Guerrero, V. M., Juárez, R. and Poncela, P. (2001) Data graduation based on statistical time series methods, *Statistics and Probability Letters*, 52: 169-175.

Guerrero, V. M. and Nieto, F. H. (1999) Temporal and contemporaneous disaggregation of multiple economic time series, *Sociedad de Estadística e Investigación Operativa (TEST)* 8, No. 2: 459-469. (Test is a Journal published by Sociedad Española de Estadística e Investigación Operativa, Madrid, Spain, and it is indexed in Current Index to Statistics)

Guerrero, V. M. and Peña, D. (2000) Linear combination of restrictions and forecasts in time series analysis, *Journal of Forecasting*, 19: 103-122.

Guerrero, V. M. and Peña, D. (2003) Combining multiple time series predictors: a useful inferential procedure, *Journal of Statistical Planning and Inference*, 116: 249-276.

Gu, C. and Wahba, G. (1993) Semiparametric analysis of variance with tensor product thin plate splines, *Journal of the Royal Statistical Society Series B*, 55: 353-368.

Haberman, S. and Renshaw, A. (1996) Generalized linear models and actuarial science, *The Statistician*, 45, 4: 407-436.

Hastie, T. and Tibshirani, R. (1990) *Generalized additive models*. London: Chapman & Hall.

Hodrick, R. and Prescott, E. (1997) Post-war U.S. business cycles: an empirical investigation*,* *Journal of Money*, Credit and Banking, 29, 1: 1-16.

Hyndman, R. J. and H. Booth (2008) Stochastic Population Forecasts Using Function Data Models for Mortality, Fertility, and Migration, *International Journal of Forecasting*, 24: 323-342.

INEGI (2005) II Conteo de población y vivienda 2005: Distrito Federal. México, Dirección General de Estadística. Available at http://www.inegi.org.mx/est/contenidos/espanol/proyectos/conteos/conteo2005/bd/consulta2005/t.asp?s=est&c=10401. Accessed on 19 February, 2009.

INEGI (2005) II Conteo de población y vivienda 2005: Estado de México. México, Dirección General de Estadística. Available a http://www.inegi.org.mx/est/contenidos/espanol/proyectos/conteos/conteo2005/bd/consulta2005/pt.asp?s=est&c=10401. Accessed on 19 February, 2009.

INEGI (1940-2000) Censos Generales de Población y Vivienda: Distrito Federal. México, Dirección General de Estadística.

INEGI (1940-2000) Censos Generales de Población y Vivienda: Estado de México. México, Dirección General de Estadística.

INEGI, Estadísticas vitales: Distrito Federal, México, 1994-2000. Available at http://www.inegi.gob.mx/inegi/default.asp/. Accessed on 30 June, 2006. (Sistema Municipal de Base de Datos (SIMBAD)).

INEGI Estadísticas vitales: Estado de México, México, 1994-2000. Available at http://www.inegi.gob.mx/inegi/default.asp/. Accessed on 30 June, 2006. SIMBAD.

INEGI (1995) I Conteo de población y vivienda 1995: Distrito Federal. México, Dirección

General de Estadística, Available at http://www.inegi.gob.mx/inegi/default.asp/. Accessed on 8 December, 2006. SIMBAD.

INEGI (1995) I Conteo de población y vivienda 1995: Estado de México. México, Dirección General de Estadística, Available at http://www.inegi.gob.mx/inegi/default.asp/. Accessed on 8 December, 2006. SIMBAD.

Jeon, Y. and Shields, M. (2008) The Impact of Relative Cohort Size on U.S. Fertility, 1913-2001, IZA *Discussion Papers* 3587, Institute for the Study of Labor (IZA).

Kauermann, G. (2005) A note on smoothing parameter selection for penalized spline smoothing, *Journal of Statistical Planning and Inference*, 127: 53-69.

Keilman, N., Pham, D. and A. Hetland (2002) Why Population Forecasts Should be Probabilistic – Illustrated by the Case of Norway, *Demographic Research*, 6: 409-454.

King, R. and Rebelo, S. (1993) Low frequency filtering and real business cycles. *Journal of Economic Dynamics and Control*, 17: 207–231.

Kitagawa, G. and Gersch, G. (1996) *Smoothness Priors Analysis of Time Series*. New York: Springer-Verlag.

Land, K. and Cantor, D. (1983) ARIMA Models of Seasonal Variation in U.S. Birth and Death Rates, *Demography*, 20: 541-568.

Laporte, A. and Fergusonb, B. (2003) Income inequality and mortality: time series evidence from Canada, Journal of infection and Public Health, 66, 1:107-117.

Laxton, D. and Tetlow, R. (1992) A simple multivariate filter for the measurement of potential output, *Bank of Canada Technical Report* 59, Bank of Canada.

Lee, R. (1992) Stochastic demographic forecasting, *Population Forecasting*, 8, 3: 315–327.

Lee, R. (1993) Modeling and forecasting the time series of US fertility: Age patterns, range, and ultimate level, *International Journal of Forecasting*, 9: 187–202.

Lee, R. and Carter, L. (1992) Modeling and Forecasting U.S. Mortality, *Journal of the American Statistical Association*, 87, 419: 659-675.

Lee, R. and Tuljapurkar, S. (1994) Stochastic population projections for the United States: Beyond high, medium and low, *Journal of the American Statistical Association*, 89, 428: 1175–1189.

Lütkepohl, H. (2005) *New introduction to multiple time series analysis.* Springer, New York.

McNown, R. and Rajbhandary, S. (2003) Time series analysis of fertility and female labor market behavior, *Journal of Population Economics*, Springer, 16, 3: 501-523.

McNown, R. and Ridao-Cano, C. (2005) A time series model of fertility and female labour supply in the UK, *Journal Applied Economics*, 37: 521–532.

McNown, R. and Rogers, A. (1989a) Forecasting mortality: A parameterized time series approach, *Demography*, 26, 4: 645–660.

McNown, R and Rogers, A. (1989b) Time-series analysis forecasts of a parameterised mortality schedule, 107-123 in P. Congdon and P. Batey (eds.) Advances in regional demography: information, forecasts, models. London: Belhaven Press.

McNown, R. and Rogers, A. (1992) Forecasting Cause-Specific Mortality Using Time Series Methods. *International Journal of Forecasting*, 8:413-432.

Manton, K., Patrick, C. and Stallard, E. (1980) Mortality model based on delays in progression of chronic diseases: Alternative to cause elimination model, *Public Health Reports*, 95: 580-588.

Nieto, F. H. (2007) Ex Post and Ex Ante Prediction of unobservable Muliivariate Time Series: A Structural-Model Based Approach*, Journal of Forecasting*, 26: 53-76.

Nieto F. H. and Guerrero V. M. (1995) Kalman filter for singular and conditional state-space models when the system state and the observational error are correlated, *Statistics and Probability Letters*, 22: 303-310.

Oeppen, J. and Vaupel, J. (2002) Broken limits to life expectancy, *Science*, 296, 5570: 1029-1031.

Okita, Y., Pfau, W. and Thanh, G (2009) A Stochastic Forecast Model For Japan's Population, Discussion Paper:09-06, JEL Classification: J1, C53.

Olshansky, S. (1987) Simultaneous/multiple cause-delay (SIMCAD): An epidemiological approach to projecting mortality, *Journal of Gerontology*, 42: 358-365.

Olshansky, S. (1988) On forecasting mortality, *Milbank Quarterly*, 66: 482-530.

Olshansky, S., Carnes, B. and Cassel, C. (1990) In search of Methuselah: Estimating the upper limits to human longevity, *Science*, 250: 634-640.

Olshansky, S., Carnes, B. and Désesquelles, A. (2001) Prospects for Human Longevity in an Aging World, *Science*, 291: 1491–1492.

Ortega, A. (2003) La paleodemografía: ¿un instrumento para simular el comportamiento demográfico del pasado? Análisis comparativo con la demografía histórica en la Ciudad de México del siglo XIX, *Estudios Demográficos y Urbanos*, 19, 1: 181-214.

Pankratz, A. (1989) Time series forecasts and extra-model information, *Journal of Forecasting*, 8: 75-83.

Pavía, J. (2000) Desagregación conjunta de series anuales: perturbaciones AR(1) multivariante, *Investigaciones económicas*, 24, 3: 727-737.

Pawitan, Y. (2001) *In All Likelihood: Statistical Modelling Inference Using Likelihood*. Oxford Science Publications.

Proietti, T. (2005) Forecasting and Signal Extraction with Misspecified Models, *Journal of Forecasting* 24: 539-556.

Rogers, A., McNown, R. and Knudsen, C. (1993) Forecasting fertility: an application of time series methods to parameterized model schedules, *Social Science Research*, 22: 1-23.

Rossi, N. (1982) A note on the estimation of disaggregate time series when the aggregate is known, *The Review of Economics and Statistics*, 64: 695-696.

Ruppert, D. (2002) Selecting the Number of Knots for Penalized Splines, *Journal of Computational and Graphical Statistics*, 11: 735-757.

SS (1993) Compendio histórico: Estadísticas vitales 1893-1993, Distrito Federal. Dirección General de Estadística Informática y Evaluación, México.

SS (1993) Compendio histórico: Estadísticas vitales 1893-1993, Estado de México. Dirección General de Estadística Informática y Evaluación, México.

Taylor, J., Siqueira, A. and Weiss, R. (1996) The Cost of Adding Parameters to a Model, *Journal of the Royal Statistical Society B*, 58: 593-607.

Theil, H. (1963) On the use of incomplete prior information in regression analysis, *Journal of the American Statistical Association*, 58: 401-414.

Thompson, P., Bell, W., Long, J. and Miller R. (1989) Multivariate time series projections of parametrized age-specific fertility rates, *Journal of the American Statistical Association*, 84: 689-699.

Tuljapurkar, Shripad, Ronald Lee, and Qi Li. (2004) Random scenario forecasts versus stochastic forecasts, *International Statistical Review*, 72, 2: 185–199.

Wei, W. W. (1990) *Time series analysis*. California: Addison-Wesley, Publishing Company, Inc..

Wood, S. (2003) Thin plate regression splines, *Journal of the Royal Statistical Society Series B*, 65: 95-114.